DOCUMENT RESUME

ED 454 837 IR 020 816

TITLE	ICCE/ICCAI 2000 Full & Short Papers (System Design and Development).
PUB DATE	2000-11-00
NOTE	235p.; In: Learning Societies in the New Millennium: Creativity, Caring & Commitments. International Conference on Computers in Education/International Conference on Computer-Assisted Instruction (Taipei, Taiwan, November 21-24, 2000); see IR 020 792. Some figures contain very small and illegible type.
PUB TYPE	Collected Works - General (020) -- Speeches/Meeting Papers (150)
EDRS PRICE	MF01/PC10 Plus Postage.
DESCRIPTORS	*Computer Uses in Education; *Educational Technology; Elementary Secondary Education; Foreign Countries; Higher Education; *Instructional Design; *Systems Development; World Wide Web
IDENTIFIERS	*Learning Environments; *Web Based Instruction

ABSTRACT

This document contains the full and short papers on system design and development from ICCE/ICCAI 2000 (International Conference on Computers in Education/International Conference on Computer-Assisted Instruction) covering the following topics: a code restructuring tool to help scaffold novice programmers; a framework for Internet-based distributed learning; a methodology for learning pattern analysis from World Wide Web logs; a novel distance learning system; an adaptive learning environment framework; an educational system that can visualize behavior of programs on the domain world; a multimedia composition-making system for children; a campus distance learning system using Multicast; the multi-tier architecture of component-oriented multimedia CAI (Computer-Assisted Instruction) systems on the Internet; CAI system generator on the Web using automatic trace recording; CoCoAJ (Communicative Collection Assisting System for Java) supporting online correction of hypermedia documents for CALL (Computer-Assisted Language Learning); an in-service training Web site for school teachers; a content analysis of journal articles and implications for instructional technology; a Chinese Web-mail system; an N-tiered heterogeneous virtual school administration system; a Web-based school official memorandum system; teaching models in Web-based teacher training; design of multiple metaphors in user interface; a Web-based action learning environment; a Web-based learning environment for overseas Chinese education; a Web-based language learning environment; 3D simulation programs for classical mechanics; a Web system to support computer exercises; a CAI system with character code discrimination on a Web environment; an intelligent learning support system with a large knowledge base; an object-oriented learning environment based on XML (eXtensible Markup Language); integrating information technology with language pedagogy for a second language online writing system; an Internet video on demand system of classroom teaching cases; knowledge analysis of tasks for courseware design; making the most of the Internet's potential for education; natural language-like knowledge representation for multimedia educational systems; an XML-based knowledge sharing and management system supporting research activities; real-time handwriting communication systems for distance education and collaborative

Reproductions supplied by EDRS are the best that can be made
from the original document.

learning; application of uncertainty reasoning for an intelligent tutoring system; automatic construction of teaching materials from course outlines; a learning support system for converting Web pages; estimation of music genres using neural network and its educational use; usability aspects of a universal brokerage and delivery system for Pan-European higher education; use of abstraction levels in the design of intelligent tutoring systems; using highly sophisticated middleware for building arbitrarily distributed teaching environments; and Xtrain--a GUI (Graphical User Interface)-based tool for multimedia presentations instruction, and research. (MES)

Reproductions supplied by EDRS are the best that can be made
from the original document.

ED 454 837

ICCE/ICCAI Full & Short Papers (System Design and Development)

PERMISSION TO REPRODUCE AND
DISSEMINATE THIS MATERIAL HAS
BEEN GRANTED BY

G.H. Marks

TO THE EDUCATIONAL RESOURCES
INFORMATION CENTER (ERIC)

1

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)

☑ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

IR020816

BEST COPY AVAILABLE

HOT NEWS
Background
Committees
Call for Papers
Deadlines
Registration
Program
Paper Acceptance
Reviewers
Venue & Travel Info.
Contact Info.
Host & Sponsors
Related Websites
Discussion Board
Proceedings
HOME

Proceedings

Content

▸Full & Short Papers (System Design and Development)

A Code Restructuring Tool to help Scaffold Novice Programmers
A Framework for Internet Based Distributed Learning
A Methodology for Learning Pattern Analysis from Web Logs by Interpreting Web Page Contents
A Novel Distance Learning System for the TIDE Project.
Adaptive Learning Environment Framework
An Educational System that can Visualize Behavior of Programs on the Domain World
An Empirical Study of Design and Use of a Multimedia Composition-Making System for Children
An Implementation of Campus Distance Learning System Using Multicast
Building the Multi-tier Architecture of Component-Oriented Multimedia CAI Systems on Internet
CAI system generator on web -- using automatic trace recording
CoCoAJ: Supporting Online Correction of Hypermedia Documents for CALL
Construct in-service Training Web Site for School Teachers
Defining Educational R&D: A Content Analysis of Journal Articles and Implications for Instructional Technology
Design and Implementation of a Chinese web-mail system
Design and Implementation of A N-Tiered Heterogeneous Virtual School Adminstration System
Design and Implementation of a WWW-Based School Official Memorandum System
Design and Implementation of Teaching Models in Web-based Teacher Training
Design of Multiple Metaphors in User-Interface
Designing a Web-Based Action Learning Environment -Integrating Learning and Working in One Environment
Developing an Effective Web-Based Learning Environment for Overseas Chinese Education
Developing Web-Based Language Learning Environment
Development of 3D simulation programs for classical mechanics - Using Java 3D -
Development of a Web System to Support Computer Exercises and its Operation
Development of CAI System with Character Code Discrimination on WWW Environment

BEST COPY AVAILABLE

Development of Intelligent Learning Support System with Large Knowledge Base
Development of the Web-based classroom system which a teacher can apply
Implementation of An Object-Oriented Learning Environment Based on XML
Integrating Information Technology with Language Pedagogy for a Second Language Online Writing System
Internet Video on Demand System of Classroom Teaching Cases
Knowledge Analysis of Tasks for Courseware Design
Making the Most of the Internets Potential for Education
Natural language-like knowledge representation for multimedia educational systems
Proposal of an XML-based Knowledge Sharing and Management system supporting Research Activities
Real-Time Handwriting Communication Systems for Distance Education and Collaborative Learning
The Application of Uncertainty Reasoning for an Intelligent Tutoring System
The Automated Teaching Assistant: Automatic construction of teaching materials from course outlines
The Development and Evaluation of a Learning Support System for Converting Web Pages
The Estimation of Music Genres Using Neural Network and its Educational Use
The usability aspects of an universal brockerage and delivery system for the Pan-European higher education
Use of abstraction levels in the design of intelligent tutoring systems
Using Highly Sophisticated Middleware For Building Arbitrarily Distributed Teaching Environment
Xtrain: A GUI based tool for Multimedia Presentations, Instruction, and Research

▲ HOME

A Code Restructuring Tool to help Scaffold Novice Programmers

Stuart Garner
Edith Cowan University
Pearson St., Churchlands 6018, Western Australia
E-mail: s.garner@ecu.edu.au

This paper concerns a new software tool called CORT (code restructuring tool) that has been developed by the author to help students learn programming. The paper begins by discussing the difficulties that students face when learning to program and the use of part complete solutions as a teaching and learning method that reduces the cognitive load that students experience.

CORT has been developed to support this use of part complete solutions and its features are outlined. When used by a student, a part complete solution to a given programming problem is displayed in one window and possible lines of code that can be used to complete the solution are displayed within another window. The lines can easily be moved between the windows in order to complete the solution and the solution then transferred to the target programming environment for testing purposes.

Finally, the use of CORT with both undergraduate and postgraduate students at Edith Cowan University is described, preliminary feedback from students indicating that CORT is easy to use and that they perceive that it is helping them in their learning of programming. Four different methods of using CORT have been identified and these will be the subject of future research.

Keywords: **Scaffolding, Programming, Flexible Learning**.

1 Introduction

Learning to write computer programs is not easy [3, 18] and this is reflected in the low levels of achievement experienced by many students in first programming courses. For example, Perkins, Schwartz et al [17] state that:

> *Students with a semester or more of instruction often display remarkable naivete about the language that they have been studying and often prove unable to manage dismayingly simple programming problems.*

and King, Feltham et al [8] state that:

> *even after two years of study, many students had only a rudimentary understanding of programming*

Over the years since the advent of high level programming languages in the 1960s, much has been written about the problems that students have in learning programming and many ideas and initiatives have been put forward for improvements in the teaching and learning process with varying degrees of success. In practice, the ways in which teaching and learning takes place in the domain of programming have changed little and many students still find the learning of programming a very difficult process. The challenge of learning programming in introductory courses lies in simultaneously learning: general problem solving skills; algorithm design; program design; a programming language in which to implement algorithms as programs; and an environment to support the program design and implementation [6]. In addition, students need to learn testing and debugging techniques to validate programs and to identify and fix problems that they may have within their programs.

Additionally, we are moving ever more rapidly to use more student centred and flexible learning methods within the teaching and learning process. This means that our instructional design for programming courses needs to take notice of these moves and utilise these methods. Fortunately technological improvements have also been significant over the last few years enabling us to more easily produce engaging courseware that can help students studying in a flexible learning mode. As courseware designers, we can produce electronic scaffolds to help students in their learning processes when they are studying on their own with limited access to a human tutor.

2 Use of Worked Examples in the Teaching and Learning of Problem Solving and Programming

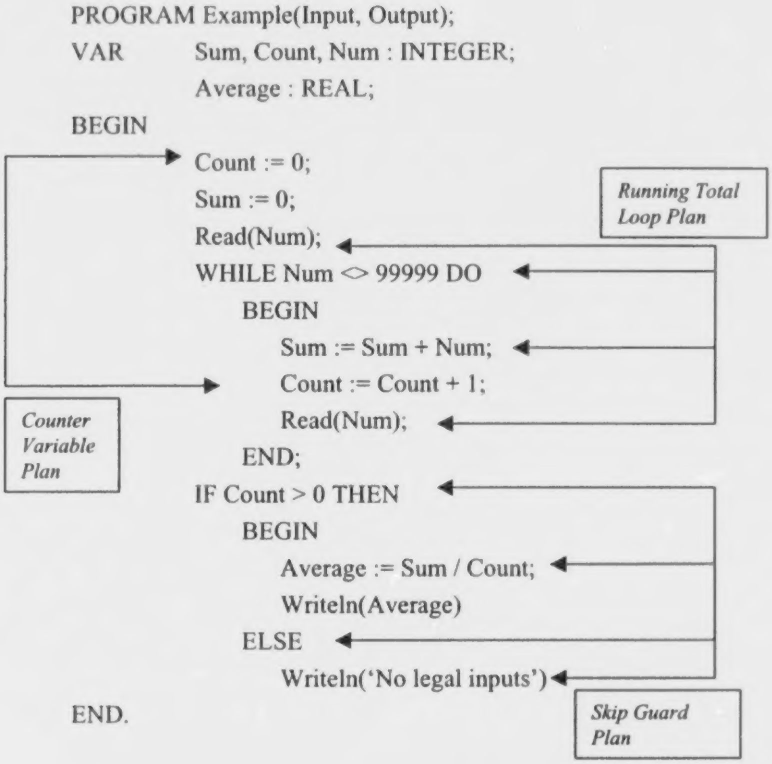
There are several methods used in the teaching and learning of programming and one of these is to utilise worked examples. Several researchers have experimented with the use of worked examples in place of conventional instruction and found strong advantages. In the domain of algebra, Sweller and Cooper [19] suggested that students would learn better by studying worked examples until they had "mastered" them rather than attempting to solve problems as soon as they had been presented with, or familiarised themselves, with new material. In their research, students studied worked examples and teachers answered any questions that the students had. Students then had to explain the goal of each problem together with the steps involved in the solution and then complete similar problems until they could be solved without errors. Sweller and Cooper found that this method was less time-consuming than the conventional practice-based model and that students made fewer errors in solving similar problems than students who were exposed to the conventional practice-based model of instruction. There was no significant difference between the "worked example" group and the "conventional" problem solving group when they attempted to solve novel problems and it was therefore concluded that learning was more efficient and yet no less effective when this worked example method was used.

Worked examples are heavily used within the “reading” method of learning programming. According to Van Merrienboer et al [22, 23] the reading approach emphasises the reading, comprehension, modification and amplification of non-trivial, well-designed working programs. However, they also suggest that presenting worked examples to students is not sufficient as the students may not “abstract” the programming plans from them, a plan being a stereotyped sequence of computer instructions as shown in figure 1.

“Mindful” abstraction of plans is required by the voluntary investment of effort and the question then arises as to how we can get students to study the worked examples properly. In practice, students tend to rush through the examples, even if they have been asked to trace them in a debugger, as they often believe that they are only making progress in their learning when they are attempting to solve problems.

Lieberman [10] suggests that students should annotate worked examples with information about what they do or what they illustrate. Another suggestion is to use incomplete, well-structured and understandable program examples that require students to generate the missing code or “complete” the examples. This latter approach forces students to study the incomplete examples as it would not be possible for their completion without a thorough understanding of the examples’ workings. An important aspect is that the incomplete examples are carefully designed as they have to contain enough “clues” in the code to guide the students in their completion. It is suggested that this method facilitates both automation, students having blueprints available for mapping to new problem situations, and schemata acquisition as they are forced to mindfully abstract these from the incomplete programs [24].

In one study, two groups of 28 and 29 high-school students from grades 10 to 12 participated in a ten lesson programming course using a subset of COMAL-80 [24]. One group, the “generation” group, followed a conventional approach to the learning of programming that emphasised the design and coding of new programs. The other group, the “completion” group, followed an approach that emphasised the modification and extension of existing programs. It was found that the completion group was better than the generation group in constructing new programs. It was found that the percentage of correctly coded lines was greater and that looping structures were more often combined with correct variable initialisation before a loop together with the correct use of counters and accumulators within the loop. It would appear that the completion strategy had indeed resulted in superior schemata formation for those students within that group. In addition, the completion group used superior comments in connection with the scope and goals of the programs, indicating that they had developed better high-level templates or schemata. It was noted in the study however that both groups were equal in their ability to interpret programs and that this might indicate that students in the completion group do not understand their acquired templates. It is then suggested that future completion strategies should include the annotation of the examples by students with details of what they are supposed to do and details of the templates (plans) that are being used.



```
PROGRAM Example(Input, Output);
VAR       Sum, Count, Num : INTEGER;
          Average : REAL;
BEGIN
          Count := 0;
          Sum := 0;
          Read(Num);
          WHILE Num <> 99999 DO
               BEGIN
                    Sum := Sum + Num;
                    Count := Count + 1;
                    Read(Num);
               END;
          IF Count > 0 THEN
               BEGIN
                    Average := Sum / Count;
                    Writeln(Average)
               ELSE
                    Writeln('No legal inputs')
END.
```

Figure 1

A side effect of the research was also noted. The drop-out rate from the completion group was found to be lower than for the generation group, particularly for female students with low prior knowledge. It was suggested that perhaps the generation of complete programs is perceived as a difficult and menacing task and that the completion strategy overcomes this difficulty.

The stimulation of the "mindful of abstraction" of schemata in students can possibly be improved further requiring them to also annotate the solutions with details of the scope and goals of the solutions and to answer questions on the inner workings of the solutions. The "degree" of completion of the solutions is an important aspect within the completion strategy and in some later work [23] examples are given of completion assignments that might be used early and later in a programming course. In an early part of a course, an example may indeed be complete and include explanations and a question on its inner workings. In the latter part of a course, the example may be largely incomplete and include a question on its workings and an instructional task. Between these two extremes, examples will have varying degree of completeness and in all cases, the incomplete examples are acting as scaffolds for the students.

3 The Cloze Procedure

A scaffolding tool called CORT (Code Restructuring Tool) has been produced that allows students to fill in lines of missing code from programs and this method is based upon the cloze procedure. The term is derived from "closure", a Gestalt psychology term referring to the human tendency to complete a familiar but not quite finished pattern [2]. The use of cloze was first used to measure comprehension in English readability [9] however it has also been used in the teaching and learning of programming as a way of measuring student understanding of programs [7, 20]. Such program comprehension tests are constructed by replacing some of the "words" or tokens by blanks and requiring students to fill in the blanks during a test. The use of the cloze procedure in testing was found to correlate well with conventional comprehension, question – answer, type quizzes and is also much easier to create and administer, see for example the work of Cook, Bregar et al [2].

Other researchers have experimented with the testing of program comprehension by omitting complete lines of code from programs and requiring students to fill in those lines [5, 13, 14, 15, 16]. Norcio found that students were more likely to supply correct statements if they had been omitted within a logic segment rather than from the beginning of a segment. This is consistent with the chunking hypothesis [12] that specifies that the first element of a chunk provides the key to the contents of the entire unit. Ehrlich looked at the differences between experts and novices in filling in missing lines within various programming plans and, as expected, found that the experts filled in the lines correctly taking into account the surrounding plan whereas novices had more difficulty.

In the various experiments in program comprehension using the cloze procedure, the students had to fill in the lines of code without being given a selection of lines to choose from. In some work done in an area unrelated to programming, students were expected to create an essay using a file of statements, only some of which were relevant to the topic [4]. The students were expected to copy and paste only the statements which they believed to be relevant and then to link them with their own text and it was suggested that learners would consolidate their understanding of the topics by having to actively evaluate all possible statements. The file of statements was acting as a scaffold to student learning.

Although the literature suggests that the cloze procedure has only been used in measuring program comprehension, it appears that it could prove useful as a way of scaffolding student learning of programming. An incomplete solution to a programming problem could be given to a student together with a choice of statements that might be used in the solution. The student would then have to study the incomplete solution and the choice of statements and decide which statements to use and where to put them. CORT uses this idea making the mechanics of placing the statements into the incomplete solution very straightforward for the student and eliminating typing errors and therefore also syntax errors.

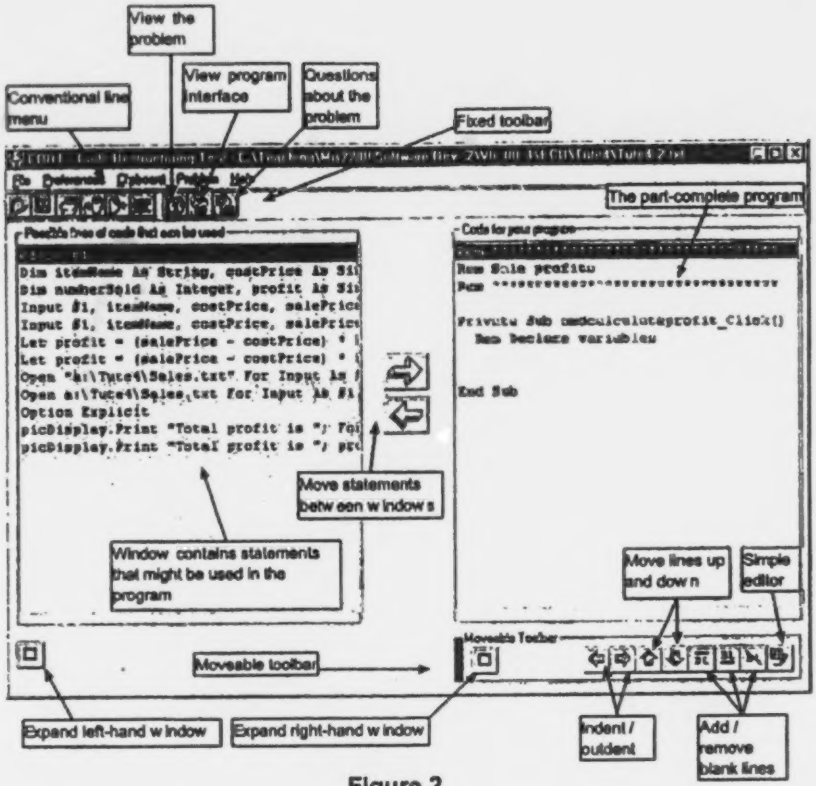
4 The Code Restructuring Tool (CORT)

CORT has been designed to support the "completion" method of learning to program and it was decided that the following features would be required in the first prototype:

- Support for part-complete solutions to programming problems. Such solutions help in schemata creation and also reduce cognitive load.
- A mechanism so that missing statements can easily be inserted into a part-complete solution and also moved within that solution. This provides scaffolding for students.
- A facility so that students can add and amend lines of code. This would allow scaffolding to be reduced and for students to add more of their own code.
- For visual programming, a facility for students to easily view the target interface. The interface should be annotated with the various object names thereby reducing any split-attention effect and helping reduce cognitive load [1].
- A facility to access tutor created questions concerning the programming problems being attempted and for students to enter answers to those questions. This will promote reflection and higher order thinking.
- A facility to easily transfer a completed solution from CORT to the target programming environment.
- A facility to easily transfer programming code from the target programming environment back into CORT for further amendment.

4.1 The CORT Design

The user interface of CORT has been designed taking into consideration the three issues that have been suggested by Marcus [11] as being fundamental to interface design, namely development, usability, and acceptance. The interface for CORT is shown in figure 2.


View the problem
View program interface
Questions about the problem
Conventional line menu
Fixed toolbar
The part-complete program
Possible lines of code that can be used
Code for your program
Move statements between windows
Window contains statements that might be used in the program
Move lines up and down
Simple editor
Moveable toolbar
Moveable Toolbar
Expand left-hand window
Expand right-hand window
Indent / outdent
Add / remove blank lines

```
Dim itemName As String, costPrice As Si
Dim numberSold As Integer, profit As Si
Input #1, itemName, costPrice, salePrice
Input #1, itemName, costPrice, salePrice
Let profit = (salePrice - costPrice) * 
Let profit = (salePrice - costPrice) * 
Open "a:\Tute4\Sales.txt" For Input As 
Open a:\Tute4\Sales.txt For Input As #1
Option Explicit
picDisplay.Print "Total profit is "; Fo
picDisplay.Print "Total profit is "; pr
```

```
Rem Sale profits
Rem ***************************************

Private Sub cmdcalculateprofit_Click()
  Rem Declare variables


End Sub
```

Figure 2

The ways in which the CORT design supports the list of required features are described in the following table.

Feature	Support in CORT Design
Support for part-complete solutions to programming problems	The part-complete solutions are automatically loaded into the right hand window and possible statements into the left hand window. Students load these from a file.
A mechanism so that missing statements can easily be inserted into a part-complete solution and also moved within that solution	Two buttons in the middle of the screen will move lines between the windows. One line, or several lines can be selected and moved across.
A facility so that students can add and amend lines of code	A simple editor is provided so that students can add their own lines or amend existing lines.
For visual programming, a facility for students to easily view the target interface	Access to this feature is via a button on the fixed toolbar.
A facility to access tutor created questions on the workings of the programming examples and to enter student answers	Access to this feature is via a button on the fixed toolbar. Student answers are automatically saved.
A facility to easily transfer a completed solution from CORT to the target programming environment	This is provided by a button on the main toolbar. A single click will copy the contents of the right hand window to the Windows clipboard ready for pasting into the Visual BASIC programming environment.

BEST COPY AVAILABLE

A facility to easily transfer programming code from the target programming environment back into CORT for further amendment	This is provided by a button on the main toolbar. A single click will paste the contents of the Windows clipboard into the right hand window, overwriting what is there.

4.2 Use of CORT by Students

A student would typically use CORT as follows:

1. A student loads in a CORT file and the two windows display a part-complete solution to a problem together with possible lines to be used. There is a facility available for the contents of the two windows to be printed out.
2. The student can view the problem statement and the Visual BASIC solution interface by clicking on the appropriate buttons on the fixed toolbar. The problem statement may have already been provided to the student in the form of a handout, however there is also a facility to print it from within CORT.
3. The student moves certain lines from the left hand window to the right hand window in an attempt to complete the solution. Lines can be moved up or down, and indented or outdented in the right hand window. Some problems have too many lines in the left hand window, some of those lines being incorrect.
4. If necessary, the student can invoke a simple editor to amend, add or delete lines of code.
5. The student clicks on the appropriate button to copy the contents of the right hand window to the Windows clipboard.
6. The student invokes Visual BASIC and loads the file that contains the interface for the solution. This is in effect the Visual BASIC solution to the problem without the lines of code and was created by the tutor.
7. The student pastes the contents of the Windows Clipboard into the Visual BASIC editor and tests the program to determine if it works correctly. Use is made of the trace and debugging facilities of Visual BASIC. These facilities provide an insight to the workings of the notional machine.
8. If the student finds a problem with the working of the program, they can return to CORT and make the changes to the code there.
9. The student repeats steps 3 to 8 until they have a working program.
10. The student answers the tutor's questions concerning the programming problem that they have just attempted.

4.3 Initial Student Feedback

CORT has been used for one semester with both undergraduate and postgraduate students in the Faculty of Business and Public Management. The particular units are in the area of software development and the language that the students learn is Visual BASIC.

Each week the students have to undertake completion programming exercises using CORT and after each problem they were asked to comment on the use of CORT for the particular problem that they had just finished. The data was collected on-line through the Web and below are some of the comments that were received:

1. It's very helpful. I can see the interface of the program before actually running it.
2. I think CORT is a very useful tool to play around the codes. It saves me time copying and pasting.
3. Considering the increased workload as the semester progresses it is a bit of a relief that the exercises are much easier with the "fill in the gap" type format in CORT.
4. Without CORT, it's sure that I'll have a lot trouble with this particular problem, which focuses on arrays (a difficult topic). Thanks CORT...
5. CORT was useful in that the part solution helped to understand the logic of VB code
6. CORT is useful . However, I have used the unit text to try to understand the indentation format when writing the code. The directional keys are great for editing the code to meet the required format.

7. This was a challenge! I think that CORT is useful so long as I am not tempted to simply manipulate code until the program runs. If I were having to write programs from scratch I would use CORT so as to format and manipulate code and modules or sub procedures etc.

5 Conclusions

As can be seen from the above, the initial feedback on the use of CORT has been favourable. We have found that students can undertake two or three small programming problems within a one hour tutorial whereas without CORT they could only undertake one such problem. Also, without using CORT students often never manage to successfully complete their assigned problems and this certainly affected their motivation.

By using CORT, students do not have to be concerned with the design of programming interfaces which considerably reduces the cognitive load in the initial stages of learning programming. Also, the reduction of "split attention affect" by labelling all the objects with their names has been very popular with the students.

The above has described a preliminary study of the use of CORT and it has been undertaken to determine its suitability and to fine tune some of its features. CORT can be used in several ways and four distinct methods have now been identified. These will be the subject of further research. The four methods are as follows:

1. **All** of the lines that are required to complete a program are made available in the left hand window of CORT. There are no extra lines displayed in the left hand window.
2. **All** of the lines that are required to complete a program are made available in the left hand window of CORT. There are also additional lines displayed in the left hand window that are not required within the program. The extra lines are similar to the required lines, however they are incorrect and act as "red herrings".
3. **Some** of the lines that are required to complete a program are made available in the left hand window of CORT. Other lines that are required for the program completion need to be keyed in by the student.
4. **None** of the lines that are required to complete a program are made available in the left hand window of CORT. All of the lines that are required for the program completion need to be keyed in by the student.

References

[1] Chandler, P. and Sweller, J. (1991). Cognitive load theory and the format of instruction. Cognition and Instruction 8: 293-332.

[2] Cook, C., Bregar, W., et al. (1984). A Preliminary Investigation of the use of the Cloze Procedure as a Measure of program Understanding. Information Processing & Management **20**(1-2): 199-208.

[3] du Boulay, B. (1986). Some Difficulties in Learning to Program. Journal of Educational Computing Research **2**(1): 57-73.

[4] Edward, N. (1997). Development of a cost effective computer assisted learning (CAL) package to facilitate conceptual understanding. CAL97, University of Exeter, UK.

[5] Ehrlich, K. and Soloway, E. (1984). An Empirical Investigation of the Tacit Plan Knowledge in Programming. Human Factors in Compter Systems. J. Thomas and M. L. Schneider. Norwood, New Jersey, Ablex: 113-133.

[6] Fowler, W. A. L. and Fowler, R. H. (1993). A Hypertext Approach to Computer Science Education Unifying programming Principles. Journal of Multimedia and Hypermedia **2**(4): 433-441.

[7] Hall, W. E., III and Zweben, S. H. (1986). The Cloze Procedure and Software Comprehensibility Measurement. IEEE Transactions on Software Engineering **May 1986**: 608-623.

[8] King, J., Feltham, J., et al. (1994). Novice Programming in High Schools: Teacher Perceptions and New Directions. Australian Educational Computing(Sep 1994,): 17-23.

[9] Klare, G. R. (1974-75). Assessing Readability. Reading research quarterly(10): 63-102.

[10] Lieberman, H. (1986). An Example Based Environment for beginning Programmers. Instructional Science **14**(3): 277-292.

[11] Marcus, A. (1992). Graphic Design for Electronic Documents and User Interfaces. New York, ACM Press.

[12] Miller, G. A. (1956). The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity to Process Information. Psychological Review(63): 81-97.

[13] Norcio, A. F. (1980a). Comprehension Aids for Computer Programs. American Psychological Association Annual Meeting, Montreal.

[14] Norcio, A. F. (1980b). Human Memory Processes for Comprehending Computer Programs. Cybernetics and Society, Cambridge, Massachusetts.

[15] Norcio, A. F. (1981). Chunking and Understanding Computer Programs. Human-Machine Systems Symposium, Boston, USA.

[16] Norcio, A. F. (1982). Indentation, Documentation and Programmer Comprehension. Human Factors in Computer Systems, Gaithersburg, Maryland.

[17] Perkins, D. N., Schwartz, S., et al. (1988). Instructional Strategies for the Problems of Novice Programmers. Teaching and Learning Computer Programming: Multiple Research Perspective. R. E. Mayer, Hillsdale, NJ: Erlbaum: 153-178.

[18] Scholtz, J. and Wiedenbeck, S. (1992). The role of planning in learning a new programming language. International journal of man-machine studies **37**: 191-214.

[19] Sweller, J. and Cooper, G. A. (1985). The use of worked examples as a substitute for problem solving in learning algebra. Cognition and Instruction **2**(1): 59-89.

[20] Thomas, M. and Zweben, S. (1986). The Effects of Program-Dependent and Program-Independent Deletions on Software Cloze Tests. Empirical Studies of Programmers: 138-152.

[21] van Merrienboer, J. J. G. (1990). Strategies for Programming Instruction in High School: Program Completion vs. Program Generation. Journal of educational computing research. **6**(3): 265-

[22] van Merrienboer, J. J. G. and Krammer, H. (1987). Instructional Strategies and tactics for the design of Introductory Computer programming Courses in High School. Instructional Science **16**(3): 251-285.

[23] van Merrienboer, J. J. G., Krammer, H. P. M., et al. (1994). Automating the planning and construction of programming assignments for teaching introductory computer programming. Automating Instructional Design, Development, and Delivery (NATO ASI Series F, Vol. 119). R. D. Tennyson, Springer Verlag, Berlin.: 61-77.

[24] van Merrienboer, J. J. G. and Paas, F. (1990). Automation and Schema Acquisition in learning elementary computer programming. Computers in Human Behavior(6): 273-289.

A Framework for Internet-Based Distributed Learning

Andreas Braun* and Andreas G. Harrer**
**Andersen Consulting, Maximilianstr. 35, D-80539 München,* andreas.braun@ac.com
***Technische Universität München, Orleanstr. 34, D-81667 München,* harrer@in.tum.de

Communication technology as well as the communication infrastructure are both changing rapidly. As a consequence, systems that support web-based learning need to be adapted due to changes in technology. This paper describes a model for web-based learning with intelligent tutoring systems (ITS) that allows separation of the concrete communication from the ITSs' implementation. The resulting framework provides a technical solution to distribute any ITS over a network. The ITS SYPROS is used to illustrate how a classical ITS can be extended to a web-based tutoring system with a maximum of code-reuse. The framework may be used freely with any ITS. To accommodate the needs of various ITSs, our model supports several architectures for distributed adaptive tutoring, including the three different models described in [3]: Master-Slave, Communicating Peers and Centralized Architectures. Our main goals are: • •Make the ITS usable for a wide range of users by supporting any web browser on any operating system.• •Offer a simple, extendable and platform independent framework to ease web-based tutoring.• • Provide a solution without royalties.• •Separate the communication technology from the client and server implementation.• •Enable method invocation and parameter passing semantics over the HTTP protocol to virtually support any web browser and users behind firewalls.• •Offer an simple user accounting and user communication functionality.• •Provide a wrapper to connect to an existing ITS. The Java source code is freely available: http://www.in.tum.de/~herzog/sypros.

Keywords: **Web-Based Learning, System Design and Development, Intelligent Tutoring Systems**

1 Introduction

Classical intelligent tutoring systems (ITS) are often platform dependant and not distributed. Modern, distributed intelligent tutoring systems (DITS) provide a more attractive solution with respect to usability and platform independence. Therefore, a modern distributed infrastructure like the internet with communication techniques like CORBA or RMI is suitable. A stable, safe and extendable basis for communication and cooperative work is needed. However, technology in this area is rapidly changing on the one hand. On the other hand communication technologies like CORBA or RMI are (still) not usable with every client, browser or platform and still have several drawbacks which prevent their usability at least for some users: Macintosh users and users with old browsers or behind firewalls/proxies who also want to use secure socket factories, only to name some.

This paper discribes a model and the resulting framework to overcome such problems. We propose to address these problems by providing an API with the semantics of object oriented remote method calls over HTTP and Servlets. Further functionality that is most likely in common for any DITS (such as user accounting and identification, security and administrative functionality) is implemented and encapsulated for ease of use.

In the current version, SYPROS is an ITS in the domain of the synchronization of parallel processes with semaphores [4], a domain of programming problems.

All the typical modules of an ITS [15] like the expert module with different types of cooperating domain experts [13], the instructional module with different tutoring strategies, the student model with cognitive and motivational traits [12], and the interface module with several support facilities, are fully implemented in SYPROS.

The current version system is a classical ITS for single-user mode and is written in C for UNIX systems. The user interface is based on the X Windows system and therefore the ITS is platform dependent. There is no direct support for multiple clients and no accounting, access control or WWW support as it would be needed for a web-based group learning system, which is our ultimate goal [11]. In the current implementation the user interface is divided at function level from the `intelligence' and database functionality, but is linked to one single executable. The proposed model will provide an application interface (API) for the client and server side. The API will encapsulate various ways of communication over a network using an abstract factory pattern [2,10]. Concrete implementations for Java RMI and servlets are provided. This model is designed to be easily extendable by other means of network transportation (e.g., CORBA or even Sockets). It will include conceptional security at an eligible level. Further, various ways of interfacing to an existing ITS on the server side are given (Java native calls to C/C++ and the connectivity to shell scripts). This factory can also be easily extended. Figure 1 shows the distribution of SYPROS. The servlet proxy Server enables connection for old webbrowsers, running not necessarily on the same machine as the `SyprosServer` implementation. Two clients are connected: "**Old Webbrowser**" connects using the servlet proxy, "**New Webbrowser**" can either use servlet communication or RMI[17]/ CORBA[22] (or anything else).

This work covers two more aspects: a security discussion for the provided model with a special focus on security issues for an ITS and a usability discussion for various platforms and webbrowsers.

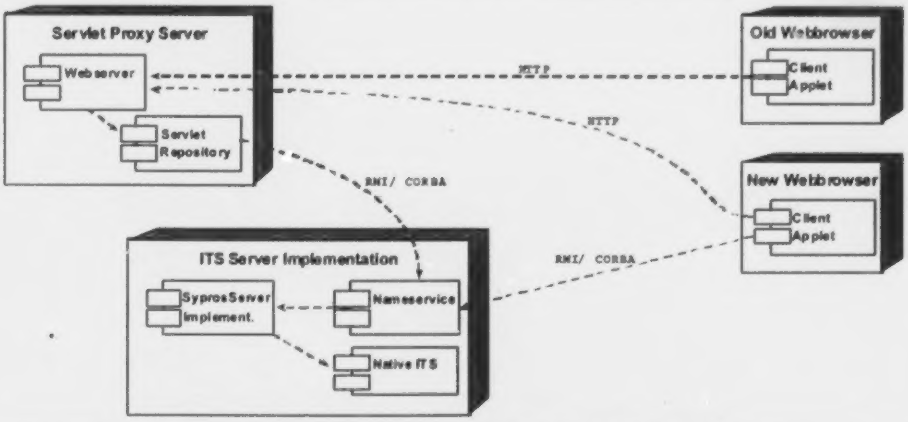


Figure 1. Distribution of client, server and servlet proxy in SYPROS (UML[19]).

Figure 2 shows the different layers for communication and levels of abstraction for a client initiated request. The dotted line between the implementation (application) level and the abstraction denotes that both the client and server implementation are separated from the underlying concrete communication. This model provides transparency in terms of process transparency. (That is, the machine on which the function or method is executed isn't known to the client's application level.) This can be compared to remote procedure calls (RPC) where the client stub and the server skeleton provide a similar transparency. In addition to that, our framework separates the concrete communication (the lowest layer in figure 2) from the application layer using the abstract communication layer. This provides transparency regarding the concrete communication technology used and therefore unburdens the application programmer from changing the application to support new technologies.

For some concrete communication implementations our framework supports language transparency as far as the client's implementation language may differ from the server's (e.g., for CORBA or Servlets).

BEST COPY AVAILABLE

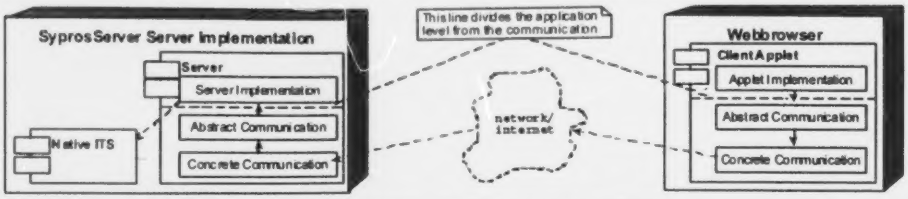

SyprosServer Server Implementation
Server
Server Implementation
Abstract Communication
Concrete Communication
Native ITS
This line divides the application level from the communication
network/ internet
Webbrowser
Client Applet
Applet Implementation
Abstract Communication
Concrete Communication

Figure 2. Levels of abstraction and layers for a request initiated by the client (UML deployment diagram).

2 Requirements

All base functionality for a distributed system is implemented. Remote method invocations are implemented independently from the Java RMI package over a `ComObject` which is JDK 1.1[9] compliant. User accounting, login procedures and access control as well as connection state information is supported directly in the framework.

A wrapper is provided to connect to an existing ITS over Java Native Interface (JNI[14]) or shell script invocation.

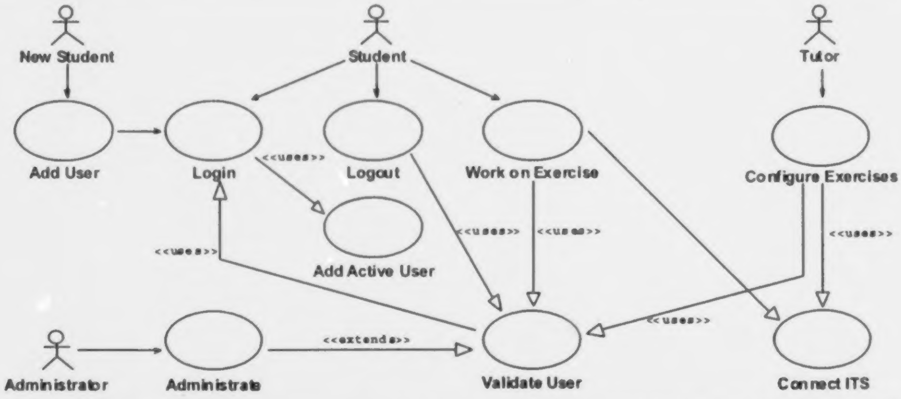

New Student
Student
Tutor
Add User
Login
Logout
Work on Exercise
Configure Exercises
<<uses>>
Add Active User
<<uses>>
<<uses>>
<<uses>>
<<uses>>
<<uses>>
Administrator
Administrate
<<extends>>
Validate User
Connect ITS

Figure 3. Some interactions among actors and use cases in SYPROS (UML use case diagram).

The use case diagram in figure 3 shows some of the use cases for SYPROS. Four types of human actors are shown in their interaction with the use cases. "**Student**" denotes an actor who is already known to the system. Therefore, "**Student**" logs into the server by passing the "**Login**" use case. "**Login**" performs authentification for which it `<<uses>>` the "**Validate User**" use case, which has knowledge of all valid user entries and so on. After accepting the user's login request some state information for that connection will be stored "**Add Active User**" and a `UserTicket` object is returned to allow stateful and secure client interaction. (`UserTicket` might be encrypted.)

"**New Student**" is an actor who is not known to the system. (Guests are handled identically.) Therefore, she can create a new user database entry herself ("**Add User**"). Later, the gathered information will be used to log into the system as described before.

An active user ("**Student**") might also use other services on the ITS server side. For example, the "**Work on Exercise**" use case first validates the call against the active users database and then uses "**Connect ITS**" (which interfaces the ITS using the wrapper) to work with the tutoring system.

"**Tutor**" is a human actor who might use the "**Configure Exercise**" use case to set up some exercises or check the student's results. The differing permissions (compared to a student) are handled by the "**Validate User**" use case.

An “**Administrator**” user will not use the client interface to connect to the server in this model. The administrator configures the databases and configuration files. Therefore, “**Administrate**” extends “**Validate User**”.

Resulting from the requirements given before our model and the framework should further satisfy the following nonfunctional requirements, pseudo requirements and design goals: The server-side installation should be simple and conceptionally platform independent. It should not be addicted to any specific web server and should work with freely available products such as Apache.

The framework is designed to be fully platform independent using the Java programming language. However, some platform dependencies exist from possible webbrowser incompatibilities and the existing ITS. In order to support old webbrowsers or users behind a firewall or proxy, a servlet repository which acts as a proxy and a servlet based client communication is provided. The communication implementation may be switched online in the client implementation[1].

The SYPROS system can be used by four groups of people: students, guests (users who are not known to the system by now), tutors and administrators (tutors who fulfill administrative functions). Therefore, the framework supports users at different level of permissions (similar to e.g., UNIX or WindowsNT).

The client applet should be small so that it is suitable even for slow modem connections. The classes needed for communication on the client side are less than 20 KB in size (without JCE security). Once the Applet is loaded, the response time of the user interface is short, as it is running locally on the client side.

The response time resulting from the security key generation and secret key exchange (Diffie-Hellman for example) of the Java Cryptography Extension (JCE) is rather long especially for strong keys and due to JCE's implementation in Java (see discussion in section 4).

Performance of the network communication depends on the underlying infrastructure. With most browsers, servlets will have a more overhead than CORBA or RMI.

The communication framework aims to support three possible client-server bindings: Static (the server name is stored in the client application), semi-static (the client locates the server once, e.g., at login time) and dynamic (the client looks up the server each time it needs to connect). Client server binding uses name resolution to find a suitable ITS server in the network. The toolkituses a server string such as **“*//{hostname|ip-address}/{service-name}*”**, just like RMI for either underlying communication infrastructure. At client implementation level, the programmer may decide whether to use static, semi-static or dynamic binding.

Together with the way of client-server binding, stateless and stateful client server connections using tickets are possible. User tickets are invented as "high-level“ stateful client-server connection for two reasons: first, the underlying ITS needs to know about the caller; tickets provide an easy way to identify the caller during a learning session. Second, encrypted ticket objects can be used to prevent attacks by intercept and replaying messages (see section 4).

Calling a remote function is somewhat dangerous if the programming language used supports *call-by-reference*[2]. For Java, *call-by-reference* is replaced by a *call-by-copy/restore* semantics. (See Java `RemoteObject` for RMI). A *call-by-copy/restore* semantics can be simulated for servlets using the `EventListener` model. In that case, the servlet proxy uses a `RemoteObject` for the server communication if the communication between the servlet proxy and the server is based on RMI and returns the object to the client using the event model. This may also be encapsulated in the framework.

In case of middleware communication such as CORBA/ RMI the *call-by-copy/restore* semantics can directly rely on the appropriate native semantics. The framework supports synchronous method invocations. Asynchronous calls can be realized using *call-by-copy/ restore*.

---

[1] This feature should be omitted for maximum compatibility with old browsers and Java engines (JDK level 1.1).

[2] Function calls that use *call-by-reference* parameter passing deliver a pointer to the value or data the parameter stores. In a distributed system with different address spaces this triggers side-effects[20].

BEST COPY AVAILABLE

Java's `try-catch`-statements are used for error handling. Therefore, the framework's error semantics is *at-most-once* by default. At application level, return values might be used to signal unexpected behavior. The SYPROS `login()`-Method for example returns `null` for the `UserTicket` if the server can't accept the login request. Although there are several possible reasons for that (e.g., unknown user, wrong password) their origin is not a communication error.

The resulting framework is described using UML notation for scenarios, use cases and object models[2,19]. The API description is given in standard Java notation[9]. The use of our framework is illustrated by the SYPROS sample.

3 A Model for the Communication Framework

Figure 4 shows the UML diagram for the SYPROS server implementation using the communication framework. The diagram shows two possible extensions for `SyprosServer`: `ComCORBA` and `ComRMI`. In the realworld implementation the programmer has to decide either to use CORBA *or* RMI, as Java does not allow multiple class inheritance.

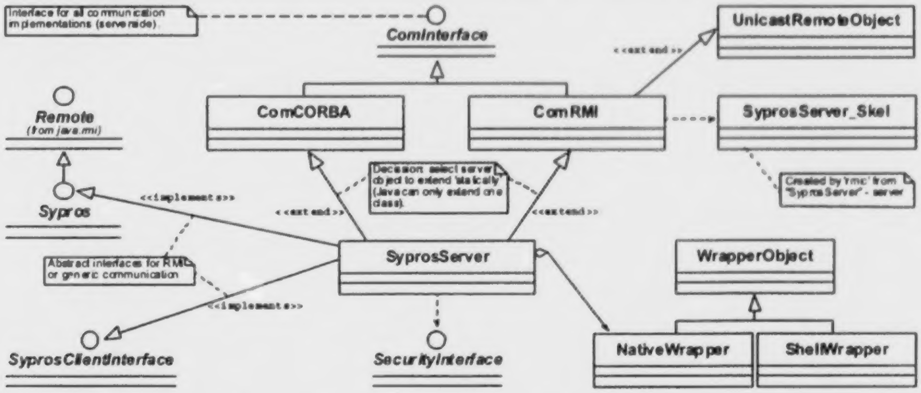

Interface for all communication implementations (serverside)
ComInterface
UnicastRemoteObject
<<extend>>
Remote
(from java.rmi)
ComCORBA
ComRMI
SyprosServer_Skel
Decision: select server object to extend statically (Java can only extend one class).
Created by 'rmic' from "SyprosServer" - server
<<implements>>
Sypros
<<extend>>
<<extend>>
SyprosServer
WrapperObject
Abstract interfaces for RMI or generic communication
<<implements>>
SyprosClientInterface
SecurityInterface
NativeWrapper
ShellWrapper

Figure 4. SYPROS server implementation. Class attributes and methods are omitted. (UML class diagram)

Therefore, there are some specialties in the server implementation: depending on the selected communication technology, the programmer has to change the head of the class definition to extend the right `ComInterface`. Further, the server has to implement the `Sypros` interface which defines the exported functions (for the RMI case). `SyprosClientInterface` contains the same definitions like `Sypros` but doesn't depend on the Java RMI classes. This ensures usability for old webbrowsers (with old Java virtual machines, VM) or clients that don't support RMI for other reasons (Macintosh).

```
import sypros.util.*;
import sypros.com.util.*;
import sypros.com.server.ComRMI;

public class SyprosServer extends} ComRMI implements Sypros {
  public  SyprosServer(String hostName, String servName) throws RemoteException
{
    super(serverHost, servName);  // create bindings
  }
  public static void main(String args[]) {
    setSecurity();                 // setup default security
  }
}
```

Figure 5. Client applet implementation for the SYPROS sample.

Figure 5 shows the class definition for the SYPROS server implementation using RMI. The underlined statements would have to be changed for a different type of communication technology.

The client implementation allows dynamic switching of the communication technology. Figure 6 shows the UML class diagram for the SYPROS client Applet. As the client communication model uses an abstract factory pattern [2,19] to create the appropriate concrete communication, the client might be a Java Applet or a Java standalone application. (The server could also be connected using the servlet URLs from HTML or other languages.)

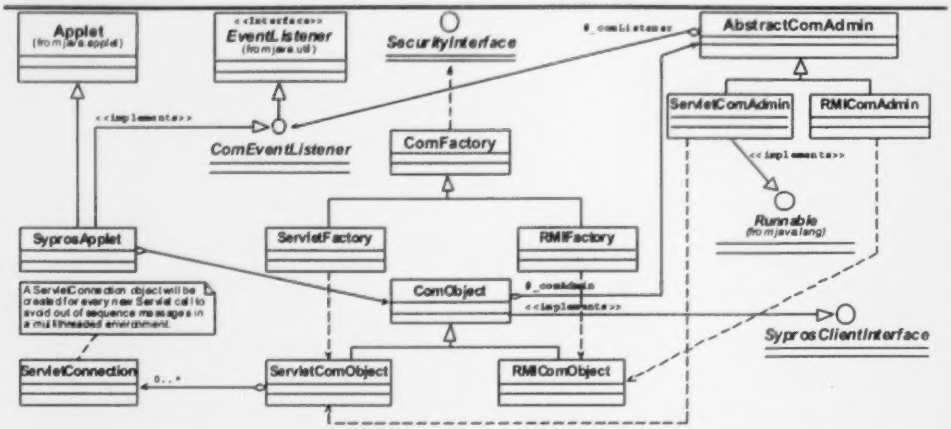

Applet
(from java.applet)
<<Interface>>
EventListener
(from java.util)
SecurityInterface
#_comListener
AbstractComAdmin
ServletComAdmin
RMIComAdmin
<<implements>>
ComEventListener
ComFactory
<<implements>>
Runnable
(from java.lang)
SyprosApplet
ServletFactory
RMIFactory
A ServletConnection object will be created for every new Servlet call to avoid out of sequence messages in a multithreaded environment.
ComObject
#_comAdmin
<<implements>>
SyprosClientInterface
ServletConnection
0..*
ServletComObject
RMIComObject

Figure 6. Client applet implementation for the SYPROS sample (UML class diagram).

The classes in the client model can be seen in three categories. First of all, SyprosApplet is the implementation for the SYPROS client interface. (Plus `Applet`, the parent.) As described before, the implementation needs not to be changed for changing communication technologies.

Then the communication classes themselves: `ComFactory`, `ComObject` and their concrete implementations provide the application interface for the implementation. `ServletConnection` is a helper that provides a per-servlet connection for persistent calls in a multithreaded application.

The `AbstractComAdmin` and its concrete implementations for servlets and RMI currently realize notification for server to client messages using the `EventListener` model and can be used for *call-by-copy/restore* type parameter passing.

4 Conceptual Security

Any internet-based application requires a special focus on security issues. The history of designing secure systems, however, teaches the inadequacy of enhancing existing systems with additional security functionality [8]. To integrate the security functionality for secure web-based tutoring, we included security policies in the framework with a top-down approach. We start by specifying the security requirements as part of the security policy:

Authentification:	All subjects and objects of the system have to be authentificated.
Total access control:	Every access to protected units has to be supervised.
Non repudiation:	Every action performed by a subject can be assigned to it's originator.
Communication privacy:	Dataflow over unsafe networks has to be adequately encrypted.
Availability:	Denial-of-Service attacks should be identified.

To meet the authentification, total access control and no-denial requirements, the framework offers integrated functionality that can be adapted or extended to your application needs. Communication privacy is provided using encrypted transmission (encrypted object serialization) based on the Java Cryptography Extension (JCE). JCE offers secret key agreement protocols (e.g., Diffie-Hellman) and encryption (e.g., Blowfish) with variable key lengths.

Ensuring the availability of a web-based service against denial-of-service attacks is maybe the hardest task. The Servlet-Proxy allows load-balancing, where the typical communication load of an ITS application (little amounts of data, long periods of thinking, infrequent transmissions) can be used to identify attacks.

5 Conclusions and Outlook

Our framework offers an easy and extendable basis for web-based distributed tutoring. The communication technology, security and ITS-integration can be easily adapted to the specific needs of an existing ITS as well as to changing communication or security technologies without rewriting the implementation for the ITS clients or server.

User accounting and access rights deliver the basis to support groups of students. However, support for cooperative work should be included in the ITS itself, like for example in SYPROS.

6 List of tested Browsers

✓: tested ok. no: tested, but failed. –: browser/ OS combination not available for testing.

Webbrowser	Version	Win98/NT		Linux		Solaris/x86		Solaris/Sparc		Macintosh	
		Servl.	RMI	Servl.	RMI	Servl.	RMI	Servl.	RMI	Servl.	RMI
Netscape	4.04	✓	✓	–	–	✓	–	–	–	✓	no
Netscape	4.05	✓	✓	✓	✓	–	–	✓	✓	-	no
Netscape	4.7	✓	✓	✓	✓	✓	✓	–	–	-	no
Netscape	4.72	✓	✓	–	–	–	–	–	–	✓	no
I-Explorer	4.0	✓	no	–	–	–	–	–	–	–	–
I-Explorer	5.0[3]	✓	✓	–	–	–	–	–	–	–	–
Java Plugin	1.2	✓	✓	✓	✓	✓	✓	✓	✓	–	–

References

[1] Braun, A., "A Client-Server Model for an Intelligent Tutoring System" (transl.), Technische Universität München, Diploma Thesis, May 2000.

[2] Brügge, B., Dutoit, A.H., "Object-Oriented Software Engineering: Conquering Complex and Changing Systems", Prentice Hall, Upper Saddle River, NJ, Oct. 1999.

[3] Brusilovsky, P., Ritter, S., Schwarz, E., "Distributed intelligent tutoring on the Web", in: Proceedings of the 8th. World Conference of the AIED Society, Kobe, Japan, Aug. 1997.

[4] Dijkstra, E.-W., "Cooperating sequential processes", in: F. Genuys (editor), Programming Languages, Academic Press, New York, Oct. 1968.

[6] Duncan Davidson, J., "Java™Servlet Specification v.2.2", Sun Microsystems Inc., Oct. 1999.

[7] Eckert, C., "Matching Security Policies to Application Needs", Proceedings of the IFIP TC11 11th Internation Conference on Information Security, Cape-Town, South Africa, May 1995, p.237-254.

[8] Eckert, C., "On Security Models", Proceedings of the IFIP TC11 12th International Conference on Information Security, Samos, Greece, May 1996.

[9] Flanagan, D., "Java in a Nutshell", O'Reilly & Associates, third edition, 1999.

---

[3] RMI-Patch applied by Microsoft Service Pack.

[10] Gamma, E., Helm, R., Johnson, R., Vlissides, J., "Design Patterns -- Elements of Reusable Object Oriented Software", Addision Wesley Publishing Company, Reading Mass., tenth edition, 1997.

[11] Harrer, A., Herzog, Ch., "SYPROS Going IDLE -- from a Classical ITS to an Intelligent Distributed Learning Environment", in: Proceedings of ICCE '99, 7th International Conference on Computers in Education, p.836-839, Sep. 1999.

[12] Harrer, A., "Both sides of the coin -- blending cognitive and motivational aspects into a tutoring strategy", in: Proceedings of the International Conference on Computers in Education, ICCE97, p.188-195, 1997.

[13] Herzog, Ch., "Cooperation and competition of syntax-oriented and execution-oriented problem solvers in intelligent programming tutors", in: Artificial Intelligence in Education -- Knowledge and Media in Learning Systems, Proc. AI--ED 97, Kobe, Japan, p.309-316, 1997.

[14] —,"Java Native Interface", JDK Documentation, Sunsoft, 1996.

[15] Martha C. Polson, Richardson, J. J., "Foundations of Intelligent Tutoring Systems", Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1988.

[16] —,"Java Remote Method Invocation -- Distributed Computing for Java", Whitepaper, http://java.sun.com/marketing/collateral/javarmi.html, Sun Microsystems Inc., 1999.

[17] —,"Java Remote Method Invocation Specification", Revision 1.50, JDK 1.2, Sun Microsystems Inc., October 1998.

[18] —,"RMI-IIOP Programmer's Guide", http://java.sun.com/products/jdk/guide/rmi-iiop/rmi_iiop_pg.html, Sun Microsystems Inc. and IBM Corporation, 1999.

[19] Rumbaugh, J., Jacobson, I., Booch, G., "The Unified Modeling Language Reference Manual", Addision Wesley Longman, Inc., Reading Mass., Dec. 1998.

[20] Tanenbaum, A. S., "Modern Operating Systems", Prentice Hall Inc., 1992.

[21] McPherson, Scott, "Java Servlets and Serialization with RMI", in: Java Developer Connection, http://java.sun.com/developer/technical/Articles/RMI/rmi/index.html, Maczieg Software, 1999.

[22] —,"The Common Object Request Broker: Architecture and Specification", Object Management Group, Wiley, New York, 1995.

A Methodology for Learning Pattern Analysis from Web Logs by Interpreting Web Page Contents

Chih-Kai Chang and Kuen-Shan Wang
Department of Information Management
Da-Yeh University
Da-Tsuen, Chang-Hwa 51505 Taiwan, R.O.C.
E-mail: chihkai@mail.dyu.edu.tw

As Web-based course become popular, the Web system accumulates a large amount of log data. Because the log data was generated by learners' behavior on the Web-based course, many researchers agree that analyzing the Web log will bring benefits for learners, instructors, and the Web site manager. In general, one record of Web log can indicate "which Web page was accessed", "who access that Web page", and "when the Web page was accessed". Although many interesting results can be derived merely depending on the general Web log, some important meanings of the Web log were not considered in previous researches. In other words, the content, represented by the Web page, is not included in the general Web log. For instance, a Web page may present homework, a discussion article, a section of curriculum, or a grade reports. However, previous research did not consider the represented content of a Web page in the Web log, in which only the file name of the accessed Web page is generally identified. This paper use data mining technology to analyze learners' online behaviors for mining learner's patterns by transforming general Web log to a content perspective. Hence, the methods of previous research still can be used to find the more meaningful results. Most important of all, our methodology finds patterns based on learning behaviors instead of browsing behaviors.

Keywords: **Web-based course, Web log, Data mining technology.**

1 Introduction

As Web-based course becomes popular, various learning activities can be running on the Web [1]. The asynchronous discussion activity, homework assignment and submission, announcement, and grade reports all can be executed on the Web. Because all the learning activities are represented as Web pages, the Web server will accumulate a large amount of log data for every Web page. Basically one record of the Web log can indicate which page was access by someone in sometime. Hence, many researches analyzed the Web server log to figure out users' motivation, users' response, browsing pattern, and the network traffic [2, 3, 4]. Furthermore, analyzing students' on-line learning behaviors and on-line problem solving activities can also discovery meaningful results [5].

There are at least 116 products of Web log analysis for commercial web sites [6]. The technologies used for analyzing Web server log evolve from traffic-based or time-based assessment to user access pattern analysis. For example, Perkowitz uses access patterns to construct an adaptive Web site [7]. Hence, the interested Web pages will be linked and organized as a proper view for every user according his/her access patterns. The *path* concept, users' sequential Web page access records, is important for constructing user access pattern for Web logs. For instance, Stuart Schechter [8] create users' path profile to predicate users' browsing behavior. Consequently, the field of Web log analysis is growing for the purpose of custom services.

Recently, applications of Web log analysis integrate data mining techniques to focus on the customer behavior patterns. It is because the predictive modeling and link analysis operations in data mining techniques can be used to answer questions such as "Which of my customs will prove to be good, long-term valuable customers and which will not?", "How can I sell more to my existing customers?", "Is there a recognizable pattern in which my customers acquire products or use services so I can market to them just-in-time?", and so on [9]. Consequently, we intuitively apply data mining techniques to Web log analysis of an instructional Web site.

For Web-based instructors, their requirements for Web log analysis differ from managers of commercial sites. One of the reasons is as Raphen Becker said, "Because many existing systems are targeted toward commercial webs, the answer is yes, course webs require different systems. One reason is simple: most instructors (and even institutions) cannot afford the commercial products, which are priced toward industry and not towards academia."[10]. Although researchers realize the differences between course webs and commercial sites, the proposed methodology for Web log analysis still inheritance from the Web logs analysis products for commercial sites. For instance, Clio project pays efforts to answer the questions such as "What are the more popular parts of the course web?", "How do readers reach particular pages?", and "Can they quickly reach the pages they want?" so on. Unfortunately, most questions of that kind can be answered by existing Web logs analysis products.

When analyzing Web logs of a course Web, we concern that one encounters what specific problems, which can not be answered by existing Web logs analysis products. In other words, only the learning characteristic of the Web-based learning environment can originate the specific problems. Our previous research focus on providing various summary report for Web instructor to solve that problems, which can not be answered by Web log analysis, from any perspectives [11]. Hence, the questions, which a instructor may ask, should be "What are the meanings of the more popular parts of the course web in learning hierarchy?", "What is the concept that leads learners to reach particular pages?', and "Can learners quickly reach the learning goals by reorganizing Web pages?' so on. In other words, the reports of existing Web logs analysis products should be interpreted to mining the pedagogical meanings by instructors, instructional designers, Web designers, and course web architects. Consequently, it is necessary to propose methodology for discovering learner (not user) access pattern in the Web-based course.

To mining the pedagogical meanings from Web logs, the first requirement is to understand the content of every Web page. In other words, the instructor of the Web course not only need to know ' who accessed the Web page', ' when the Web page is accessed', and ' from where the learner come', but also should know ' what the Web page contains'. However, it is difficult to represent the content of a Web page with symbols. The reason is that the content of a Web page may contain many concepts. Consequently, the first step for understanding the pedagogical meaning is reconstructing the Web pages in the site of a Web-based course by endowing only one topic or concept for each Web page. While breaking a Web page into single concept Web pages, one would find that some concepts are not atomic concepts. That is because a major concept will contain many sub concepts. Hence, the second step for understanding the pedagogical meaning of a Web page is to identify its location within a concept hierarchy instead of its location within the hypertext hierarchy.

The second requirement for discovering learners' learning pattern is to mining sequential access paths on previous aforementioned concept hierarchy. Although there are methodologies to reconstruct navigating paths of users' behaviors on a Web site, that information is not enough for a Web instructor to make some pedagogical decisions. The users' access (behavior) pattern can only help Web site manager improving Web site schema because a Web instructor still can not figure out learners' intention merely by analyzing Web logs without supports of the Web page content. The proposed concept hierarchy presents a feasible style for supports of interpreting the Web page content. After learners' navigating paths on a Web site are transforming to navigating paths on the concept hierarchy, a Web instructor can comprehensive how learners learn from the information of what learners read.

This paper proposes a methodology to mining learners' learning pattern by transforming learners' Web page access sequences to sequences of learning a concept in Web logs. The methodology is supported by traditional web logs mining algorithms, which is designed for discovering users' access pattern on a Web site. This methodology is not used to replace traditional web logs mining algorithms nor is arguing that concept hierarchy is a suitable web site schema. Rather, this methodology presents a framework for integrating traditional web logs mining algorithms with pedagogical meanings of web pages to support Web instructor get more feedback from learners' navigation on the Web course site. Broadly speaking, this methodology contribute to apply traditional web logs mining algorithms to a specific domain in the technical aspect and progress assessment skills in the Web-based distance learning aspect.

2 Illustrative Example

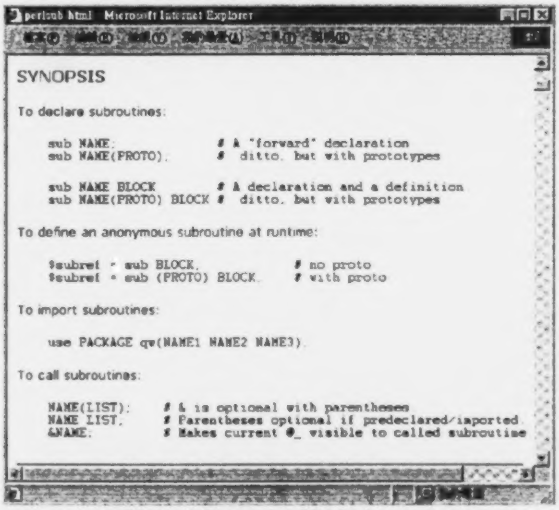
In overview, there are two steps in this illustrative example of detecting learning status. The first step is data preparation. We design a sophisticate structure of a Web site so that we can recognize the content of the accessed Web page. The second steps will find pedagogical meanings from the contents of the preferred Web pages. In this illustrative example, the result of step two will show that learner is not familiar with the learning topic.

2.1 Data Preparation

The required data was collected from the students in an undergraduate course of Perl programming. Perl is a high-level programming language written by Larry Wall. Perl is a very popular programming language for system administrators and CGI script authors. After a brief introduction of Perl, students were asked to study the Web pages extracted from Perl manual. There are three topics in the prepared Web pages. First topic of Web pages demonstrates how to execute the Perl interpreter, called *Perlrun* in Perl manual. Second topic of Web pages explains the Perl model for declaring importing, and calling a subroutine, called *Perlsub* in Perl manual. Third topic of Web pages describes associativity and precedence of Perl operators, called *Perlop* in Perl manual. Consequently, learners' behaviors recorded by Web logs can be recognized by the topic of accessing Web page.

Synopsis and description compose each topic of Web pages. Synopsis is a summary of a topic and generally contains no more than one page. Figure 1 illustrates the synopsis of the Perlsub topic. Description explains the details of a topic in original Perl manual. For illustration, description for each topic was reorganized into two Web pages. In general, synopsis of a topic is prepared for learners who are familiar with that topic. Learners who are learning a topic will prefer the description of that topic. Hence, we can help a learner just in time if he/she is always looking around the description of a topic.

Aforementioned structure is content structure of learning materials. To present learning materials in a hypertext style, a hyperlink structure is required. We use the full connection style to link all Web pages so that learners can navigate to any destination in any Web page.



SYNOPSIS

To declare subroutines:

```
    sub NAME;                 # A "forward" declaration
    sub NAME(PROTO);          #  ditto, but with prototypes

    sub NAME BLOCK            # A declaration and a definition
    sub NAME(PROTO) BLOCK #  ditto, but with prototypes
```

To define an anonymous subroutine at runtime:

```
    $subref = sub BLOCK;                 # no proto
    $subref = sub (PROTO) BLOCK;         # with proto
```

To import subroutines:

```
    use PACKAGE qw(NAME1 NAME2 NAME3);
```

To call subroutines:

```
    NAME(LIST);     # & is optional with parentheses
    NAME LIST;      # Parentheses optional if predeclared/imported
    &NAME;          # Makes current @_ visible to called subroutine
```

Figure 1. Synopsis of Perlsub.

Figure 2 shows the concept structure of the learning materials on the Web site. The notation P_i indicates the Web pages. Although the overview structure is composed of concept hierarchy and contents of learning materials without hyperlink information, the tree structure above the P_i can be used to interpret the content in the page. For instance, the P_1 belongs to concept synopsis, which is the partial content of the Perlrun topic.

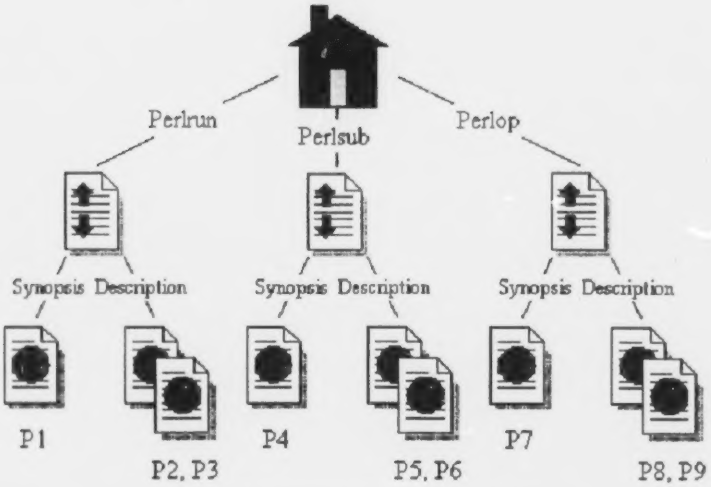


Figure 2. The concept hierarchy of a Web site.

2.2 Mining Processes

There are three learning topics in the Web site, denoted as Perlrun, Perlsub, and Perlop. Each learning topic has two sub concepts, denoted as synopsis and description. The word "synopsis" is used to indicate the Web page for summarizing a topic and the word "description" represents the Web pages that explain a topic in detail. There is an index Web page linking every Web pages to serve as communicating interface with learners. Hence, learners can study any topic in any order through the index Web page. Assume that there is a learner who prefers the "description" Web pages of any topic. In other word, that learner is not familiar with all topics. Hence, the logs of that learner' s browsing behavior on the Web site may be like the sequence:

p2, p3. p2, p8, p9. p5, p8. p5, p1. p2, p5, p6

Because learning can happen in any time, only time nearly browsing behaviors will be related in a learning pattern. Hence, the transaction idea, used in database theory, is involved to cluster learners' browsing behavior. The T*i* means a transaction of the learner' s browsing behavior.

T1: p2, p3
T2: p2, p8, p9
T3: p5, p8
T4: p5, p1
T5: p2, p5, p6

The content of every Web page can be interpreted as a pair of topic and representation style. For instance, p2 belongs to topic Perlrun and is a description of the topic. Hence, p2 is interpreted as (Perlrun, description). After interpreting the transaction data of learner' s behavior, the results are follows.

T1: (Perlrun, description), (Perlrun, description)
T2: (Perlrun, description), (Perlop, description), (Perlop, description)
T3: (Perlsub, description), (Perlop, description)
T4: (Perlsub, synopsis), (Perlrun, synopsis)
T5: (Perlrun, description), (Perlsub, description), (Perlsub, description)

Most of algorithms for mining pattern are derived from aprior [12]. We divide the problem of discovering multi-dimension learner access pattern into four sub procedures, that is litemset phase, transformation phase, sequence phase, maximal phase. Hence, we can use the aprior algorithm for mining pattern. We use the illustrative example to depict the four sub procedures. The litemset phase will generate the large-1 itemset as Table 1.

ID	Large-1 itemset	Support
1	(Perlrun, ‘ *’)	3
2	(Perlsub, ‘ *’)	3
3	(Perlop, ‘ *’)	3
4	(‘ *’, description)	4
5	(Perlrun, description)	3
6	(Perlsub, description)	3
7	(Perlop, description)	2

Table 1. Large-1 itemset.

The transformation phase use the feasible IDs of items in the large-1 itemset to substitute items in the transaction of learners' behavior. For instance, the (Perlrun, description) in T1 can be substituted by (Perlrun, ‘ *’), (‘ *’, description), or (Perlrun, description). Hence, the set of feasible IDs is {1, 4, 5}. The result after the transformation phase is following.

T1: {1, 4, 5}, {1, 4, 5}
T2: {1, 4, 5}, {3, 4, 7}, {3, 4, 7}
T3: {2, 4, 6}, {3, 4, 7}
T4: {2, 4, 6}, {1}
T5: {1, 4, 5}, {2, 4, 6}, {2, 4, 6}

The problem is simplified to mining sequential patterns after the transformation phase [13]. Consequently, the sequence phase can generate the large-2 itemset and larget-3 itemset as Table 2 and Table 3.

Large-2 itemset	Support
{1, 4}	3
{2, 4}	2
{4, 3}	2
{4, 4}	3
{4, 7}	2
{5, 4}	3
{6, 4}	2

Table 2. Large-2 itemset.

Large-3 itemset	Support
{1, 4, 4}	2
{4, 4, 4}	2
{5, 4, 4}	2

Table 3. Large-3 itemset.

Finally, the maximal phase will find the most meaningful pattern from large-2 itemsset and large-3 itemset. Initially, the union of large-2 itemset and large-3 itemset is used as the result. Then, some items will be eliminated because they are the subsets of some larger items. For instance, the meaning of {5, 4, 4} is more than its subset {5, 4} and {4, 4}. Hence, the large-2 items, {5, 4} and {4, 4}, will not be deleted from the initial result. Finally, some items will be eliminated because they are less meaningful then items in the result. For instance, the {4, 3} will be deleted because {4, 7} implies {4, 3}. Similarly, the {2, 4} will be deleted because {6, 4} implies {2, 4}. The following table illustrates the result.

Maximal itemset	Real patterns	Support
{6, 4}	{(Perlsub, description), (‘ *’ ,description)}	2
{4, 7}	{(‘ *’ , description), (Perlop, description)}	2
{5, 4, 4}	{(Perlrun, description), (‘ *’ ,description), (‘ *’ , description)}	2

Table 4. Maximal itemset.

3 Conclusion

The Web-based learning environment offers opportunities to precisely observe learning processes. However, it is tedious for a Web instructor to discovery useful information from the huge amount of Web logs. Traditionally, a Web instructor uses the Web logs analysis products to realize the unusual parts of a Web site. From the pedagogical standpoint, the results of the Web logs mining algorithms are not very useful for figure out learners' learning process because the contents of Web pages are not considered. This paper proposes a methodology to mining learners' learning pattern, which is related with the Web page contents, from Web logs. The methodology uses Web logs mining algorithms, which is used in Web logs analysis products, and the concept structure embedded in Web pages to mining patterns with pedagogical meanings, so called learning patterns. In our opinions, this methodology presents a framework for integrating traditional web logs mining algorithms with pedagogical meanings of web pages to support Web instructor figure out learners' navigation on the Web course site from the concept hierarchy perspective. Consequently, the approach presented here may be not only a feasible application of traditional web logs mining algorithms, but also a possible direction of Web-based learning assessment research.

Acknowledgements

This project is supported by the National Science Council in Taiwan under contract number NSC 89-2511-S-262-001.

References

[1] C.K. Chang and G.D. Chen. Constructing Collaborative Learning Activities for Distance CAL Systems. Journal of Computer Assisted Learning, Vol. 13, No. 1, pp. 2-15, March 1997.

[2] R. Fuller and J. de Craaff. Measuring user motivation from server log files. In http://www.microsoft.com/usability/webconf/fuller/fuller.htm

[3] T. Sullivan. Reading reader reaction: A proposal for inferential analysis of web server log files. In Proc. Third Conf. Human Factors on the Web, Denver, Colorado, June 1997.

[4] Sun, C. T. & Ching, Y. T. (1995). Hypermedia Browsing Pattern Analysis. International Journal of Education Telecommunications, 1(2-3), 293-308.

[5] Jihn-Chang J. Jehng. Exploring Learning Potentials in Network-enhanced Learning Environment. In Proc. of ED-MEDIA 99, Seattle, Washington, USA, 1999.

[6] Internet Product Watch (1999). Internet Product Watch: Analysis Products. Online document at http://ipw.internet.com/analysis/index.html (accessed March 25, 1999; no creation or modification date available).

[7] M. Perkowitz and O. Etzioni. Adaptive sites: Automatically learning from user access patterns. In Proc. 6th Int. World Wide Web Conf., Santa Clara, California, April 1997.

[8] Stuart Schechter, Murali Krishnan, Michael D. Smith, Using path profiles to predict HTTP requests. Computer Networks, (30)1-7, pp. 457-467, 1998.

[9] Peter Cabena, et al. Discovering data mining: from concept to implementation. Prentice-Hall, Inc.

[10] Raphen Becker, Kevin McLaughlin, and Samuel A. Rebelsky. Project Clio: Tools for tracking student use of course webs. In Proc. of ED-MEDIA 99, pp. 981-986.

[11] C.K. Chang, G.D. Chen, and Kou-Liang Ou. Student Portfolio Analysis by Data Cube Technology for Decision Support of Web-Based Classroom Teacher. Journal of Educational Computing Research, Vol 19(3), 307-328, 1998.

[12] Rakesh Agrawal, Heikki Mannila, Ramakrishnan Srikant, Hannu Toivonen, A. Inkeri Verkamo: Fast Discovery of Association Rules. Advances in Knowledge Discovery and Data Mining, pp. 307-328, 1996.

[13] Rakesh Agrawal and Ramakrishnan Strkant, Mining Sequential Patterns. ICDE, pp. 3-14, 1995.

BEST COPY AVAILABLE

A Novel Distance Learning System for the TIDE Project.

YAGI Keisuke* KAMEDA Yoshinari* NAKAMURA Motonori*
MINOH Michihiko* and Maha Ashour-Abdalla**
**Center for Information and Multimedia Studies, Kyoto University.*
Yoshida Nihonmatu-cho, Sakyo, Kyoto 606-8501, Japan.
Phone:+81-75-753-9048
Fax:+81-75-753-9001
E-mail:yagi@media.kyoto-u.ac.jp
***Center for Digital Innovation, University of California, Los Angeles.*
Box 951567, 3863 Slichter Hall, Los Angeles, CA 90095-1567, USA.
Phone:+1-310-825-8881
E-mail:maha@cdi.ucla.edu

A distance learning system architecture that exchanges information in real time among lecture rooms is discussed. Forty-eight functions are implemented into subsystems and used for the information exchange. Which and how many functions are implemented into each subsystem depends on system design. The distance learning system and a case study of the system design is presented. The system was implemented for the support of two undergraduate courses and was evaluated by a student survey questionnaire. Data from the questionnaire showed that the distance learning system was successfully implemented and transparent for the undergraduate users.

Keywords: **Real-time System, International Project.**

1 Introduction

Research in distance learning has primarily focused on two types of systems: storage based and real time. Real time systems have been seen to be advantageous for interactive discussions between a lecturers and students. For the purposes of this paper, the phrase "distance learning system" refers to a real time type of system. The present paper, addresses the implementation of a real time system to support the TIDE (Trans-Pacific Interactive Distance Education) project. TIDE is a collaborative project among Kyoto University, University of California Los Angeles (UCLA), and Nippon Telephone and Telegraph Corporation (NTT). System architecture, the integration of 48 functions to support the distance learning goals, and other technical issues are discussed. The implementation of the real time system in two undergraduate courses was assessed using questionnaire and survey data.

2 Distance Learning System Architecture

2.1 Information about the lecture

The term lecture is commonly defined as communication between a lecturer and students in a typical lecture hall. However, communication is not limited to that which occurs between the lecturer and the students. Thus, for the purposes of this paper, we use the term "participants" to refer to both lecturers and students. In a typical lecture hall, participants can communicate using all five senses. While audio and visual exchange is easy during a typical lecture, information exchange through smell, taste and touch are more difficult. Thus, in discussing information exchange over a distance, we focused on communication by visual and audio information exchange. Handouts, electronic documents, writing on blackboards, and other nonverbal materials are essential tools for participants involved in a lecture. Additionally, there needs to be a method where participants can point out the visual information of interest to share with other participants. In distance learning situations, pointing needs to be shared among the lecture rooms involved. Thus, in a distance learning environment, the following seven kinds of information need to be exchanged. 1) non-verbal information, 2) content of physical material, 3) content of the electronic material, 4) content of written material, 5) pointing information about the physical material, 6) pointing information about the electronic material, 7) pointing information about the writings. Verbal communication among participants is one of the most important communication exchanges. Teaching materials like videotapes make sounds. These sounds can be mixed with verbal information easily without degradation. Thus, exchanging verbal information is discussed within the framework of exchanging audio information.

2.2 Function classes for the information exchange

In order to exchange information among lecture rooms, the distance learning system needs to 1) capture, 2) encode and 3) transmit the information to the other lecture rooms. The transmitted information needs to be 4) received, 5) decoded and 6) presented in the other lecture rooms. Thus, six function classes are required for the system.

2.3 System Architecture

From the eight kinds of the information and the six function classes, forty-eight functions are required of the distance learning system. Which and how many functions are implemented into a subsystem depends on system design. The following describes the case study of the system design for the TIDE project. Fundamentally, the visual information can be captured as video streams using cameras. Thus, we initially designed a camera subsystem to implement the seven capturing functions to support visual communication. Because visual information can be presented with projectors, we designed a projection subsystem to implement the seven exposing functions of visual communication. Verbal information can be captured with microphones and exposed with speakers. The exposed sounds are captured with the microphones and causes acoustic echo. In order to cancel the echo, we have implemented the capturing and exposing functions into an audio subsystem. The visual and audio information need to be encoded at the same time for synchronization and the encoded information needs to be decoded at other lecture rooms. To address these issues, we designed a codec subsystem. For transmitting and receiving encoded information from the codec subsystem, we designed a quality of service guaranteed network subsystem. During a lecture, several video streams need to be exchanged because the talking participant and the teaching material are not always occurring in the same location. Unfortunately, our network subsystem has limited bandwidth and it allows the codec subsystem to exchange only one video stream at a time. Thus, we redesigned our system using PC based subsystems as follows: Using an electronic whiteboard, writing content can be captured as vector data. The data can be encoded and decoded on a PC. Thus, we designed an electronic whiteboard subsystem to implement capturing, encoding and decoding of the writing. Because vector data does not require the bandwidth required for audio/video streams, it is transmitted and received over the Internet. When the material on a PC is used, content and pointing information exists locally on the PC and needs to be transmitted and synced with the rest of the components. Thus, we designed a teaching material synchronizing subsystem to implement capturing, encoding and decoding whiteboard functions. The synchronizing subsystem also transmits and receives information over the Internet. Based on the discussion above, Figure 1 shows our distance learning system architecture. Each block in Figure 1 represents a subsystem.

3 Distance learning system for the TIDE project

3.1 Audio subsystem

The audio subsystem needs to address problems with acoustic echo and electronic echo. The electronic echo occurs when a received signal is mixed into the transmitting signal. In our audio subsystem, multiple audio mixers are used to separate the received signal and the transmitting signal.

3.2 Camera subsystem

The camera subsystem consists of four observation cameras and four shooting cameras. Checking the motion region in the successive two frames of video images from the observation camera detects the region of the participant on the image frame. The location of the participant is calculated by giving, in advance, the camera parameters of location, direction, focal length, image aspect, etc. After that, the appropriate camera, camera pan, tilt, and zoom, are chosen to shoot the moving object and/or speaking participant [4].

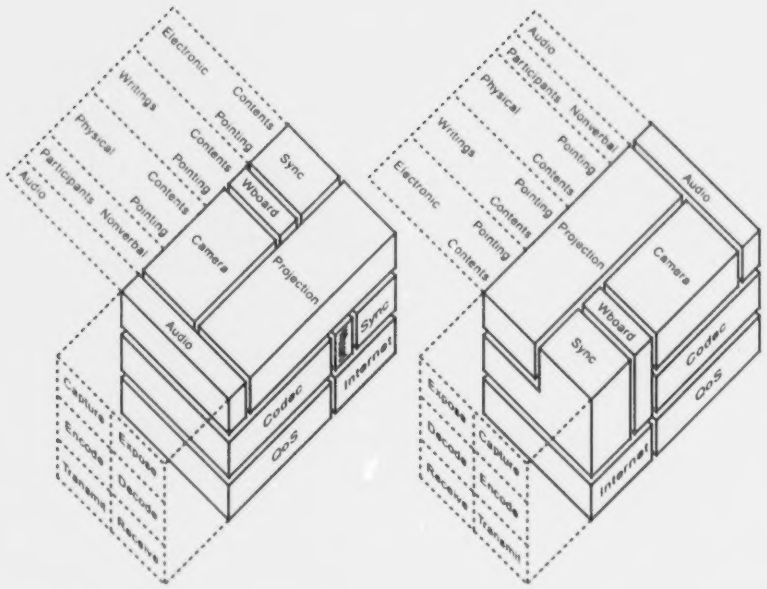

Electronic
Contents
Writings
Pointing
Contents
Physical
Pointing
Contents
Participants
Nonverbal
Audio
Pointing
Sync
Wboard
Camera
Projection
Audio
Codec
QoS
Sync
Internet
Capture
Encode
Transmit
Expose
Decode
Receive
Audio
Participants
Nonverbal
Physical
Pointing
Writings
Contents
Pointing
Electronic
Contents
Pointing
Contents
Projection
Audio
Camera
Wboard
Sync
Codec
QoS
Internet
Expose
Decode
Receive
Capture
Encode
Transmit

Figure 1: The system architecture of the distance learning system designed for the TIDE project.

3.3 Codec subsystem

For real time communication among participants, the delay caused by encoding and decoding needs to be minimized. It is very time consuming to make highly compressed data streams without degrading quality. For the purposes of this system, the MPEG2 standard was chosen for our codec subsystem. The codec encodes and decodes audio/video signal of 3Mbps in 300 msec.

3.4 Electronic whiteboard subsystem

The electronic whiteboard has two laser scanners on the top and observes position and color of writing marker. The observed position and color information is transferred from the whiteboard to the PC via serial connection. The transferred information can be easily browsed on a PC monitor. Additionally, the whiteboard information can be browsed simultaneously in multiple locations over the Internet.

3.5 Projection subsystem

The projection subsystem presents visual information on video screens. In a lecture hall, it is optimal if students are able to see the screen and the lecturer at the same time. Likewise, it is important that the lecturer also see the screen and students simultaneously. To address these issues, our projection subsystem has large screens behind the lecturer for student viewing, as well as small video monitors in front of the lectern for lecturer viewing.

3.6 Quality of service guaranteed network subsystem

For the stable transmission of the data stream, the system mainly uses ATM (Asynchronous Transfer Mode) network technology. The network subsystem is composed of three parts: GEMnet[1], CalREN2 [2] and Abilene[3]. GEMnet is an intra ATM network of NTT. CalREN2 and Abilene are parts of Internet2 in the United States. GEMnet interconnects Kyoto University and the NTT America Cupertino office over the Pacific Ocean. On the GEMnet, a PVC(Permanent Virtual Channel)connection is reserved with a guaranteed bandwidth of bi-directional CBR(Constant Bit Rate) 5Mbps.

Table 1: Regression coefficients of the principal component of the questionnaire survey.

Principal Coefficient	Middle	End
Interactivity	0.322	0.382
Quality of the teaching materials	(0.107)	0.414
Effectual manipulation of the system	0.304	(0.168)
Presence	0.561	0.637
Unstability of the system	-0.257	(-0.143)

3.7 Teaching material synchronizing subsystem

In order to present the electronic material to all lecture halls simultaneously, the synchronizing subsystem pre-loads the teaching material and transmits the mouse events. This software is appropriate for presenting dynamic teaching materials like movie files.

4 Evaluation

We have conducted two courses between Kyoto University and UCLA from October through December 1999 using our distance learning system. The distance learning system is evaluated by survey questionnaires given to the students at the middle and upon conclusion of each course[5]. The questionnaires asked students to rate various factors of the course on a scale of 1 to 5. We performed a principal component analysis of the survey and found following principal components: 1) interactivity, 2) quality of the teaching materials, 3) manipulation of the system, 4) presence and 5) instability of the system. We also performed a regression analysis to identify predictions of satisfactory grades in the courses. Table 1 shows the regression coefficients derived from the analysis. The coefficients in parentheses exceed the significance level of 5%. As the course proceeded, students' attention shifted from the novelty associated with the new technologies to the teaching materials and course content. This indicated that our system was of high enough quality to be a transparent medium for the students.

5 Conclusions

In this paper, we discussed a distance learning system architecture that exchanged information in real time among lecture halls. After identifying that there exist forty-eight functions required for information exchange, the distance learning system was introduced as a case study. We introduced the seven necessary subsystems and the technical issues surrounding the implementation of these subsystems. Our system was used for two undergraduate courses and evaluated by a survey questionnaire. The data from the questionnaire showed that our system, over time, became transparent for the students.

References

[1] http://www-grd.rdh.ecl.ntt.co.jp/GEMnet/

[2] http://www.cenic.org/salren2.html

[3] http://www.ucaid.edu/Abilene/

[4] KAMEDA Yoshinari *et. al*: "A Live Video Imaging for Multiple Users," Proc. of ICMCS'99, Vol. 2, pp.897 – 902, 1999.

[5] MURAKAMI Masayuki, YAGI Keisuke *et. al*: "Evaluation of distance learning system by the student in Japan and USA," Technical Report of IEICE, ET99-104, pp.67 - 74, Mar. 2000(Japanese).

Adaptive Learning Environment Framework

Christof Dallermassl, Heimo Haub, Harald Krottmaier,
Klaus Schmaranz, Philipp Zambelli
{cdaller,hhaub,hkrott,kschmar,pzamb}@iicm.edu

Institute for Information Processing and
Computer Supported new Media

In this paper the *Adaptive Learning Environment Framework* is presented. This Framework allows to re-use, combine, and improve existing learning systems or knowledge-databases and equip the resulting meta-learning environment with a new advanced user interface. Especially the last mentioned feature supports a completely new way to learn different materials from different sources without having to switch between systems. The introduction of an abstraction layer between different learning environments and the introduction of different views on the contained learning material allows to combine, extend, and improve available learning systems easily.

Keywords: **Personalized Learning Environment, Middleware, Application Framework, Distributed Object Systems**

1 Motivation

Different learning environments offer different ways to visualize the information and also different functionality (e.g. add annotations, communicate with other students, ...), but *views*, *functionality*, and *data* usually are tightly coupled. This implies that people using more than one learning system have to adopt to different user interfaces, a fact that does slow down learning speed a lot. As can also be seen in the classical learning environment (table, books, paper, pencil) any minor changes in the environment very often have a negative impact on the learning quality and so on the learning speed. Once the learning environment (the table) suits perfectly for a student, only the learning material (the books) and the used tools (preferred pencil, ruler, or calculator) are subject to change.

Neither of today's computer based learning environments is a perfect solution concerning transfer speed, actuality of the material or the way information is presented. Additionally, the majority of learning systems are incapable of using information prepared for other systems, because of incompatible data formats or communication protocols.

A system that allows to combine different electronic learning systems and that also allows to use the superset of their features, thus extending it with additional functionality (e.g. the ability to add annotations to the learning material), could increase the quality of the learning process enormously.

2 System Requirements

Usually learning systems are very hard to extend, as the learning material is tightly coupled to the view on the data. The classical method to overcome this problem is to split up data, view and functionality (Model-View-Controller design pattern [3]). This is also the first and most important requirement of the described framework: provide a *modular, component-based architecture* that makes it easy to create an adaptive learning environment that allows to *re-use data* from available learning systems (e.g. Gentle [4] or Dictionaries).

Separation of material, representation, and functionality offers great possibilities: It allows us to create a *superset of information*: Imagine having different learning systems available, each of them representing a specific subject. For example take a CD-ROM about World War Two, including historic images and reports and a geographical information system that publishes maps and information via Internet (Web-browser). But only a combination of these two information sources allows the users to get a deeper understanding of the facts and figures of each system.

The technique can also be used to improve existing systems and supply them with new functionality that wasn't previously foreseen. Once a module is created that is e.g. able to handle annotations or allow online discussion with other students, all participating learning systems benefit from this functionality.

Another important part is the interface the user is working with. It is at least as important as the contained material, as a good human computer interface optimizes the learning effort [7, 5]. We use the term *view* to describe the (mostly visual) appearance of the learning material. This appearance depends on various factors, e.g. the *output device* the user works with, the *environment* the user works in, the *role*, the user plays (student, administrator, instructor, ...).

According to the Model-View-Controller design, the view on the information must be decoupled from the data-model. This implies that there may exist *different views for the same content.*

As different users play different roles in connection with the learning material, different roles may have different *access rights* to the information. Even if the data source does not provide an access control system, it must be guaranteed that only users with sufficient rights may edit/change/delete the educational or administrative data.

3 Concept

The facts that the *Adaptive Learning Environment Framework* has to be modular, has to decouple data source(s) from view(s), and may add functionality to the underlying learning systems (see section 2) results in a middleware design approach: the framework is put in between the data source containing learning material and views that are used to work with the information.

The Dinopolis [2] framework, which is being developed on the IICM provides such a generic framework [1]. It allows the integration of various types of databases or applications and is able to provide the content in a highly dynamic way to a broad range of clients/views. The integration of learning material is one special application of the general concept of Dinopolis.

4 Views

A *view* is responsible for giving a visual and logical representation of the learning material. It is completely decoupled from the actual information, so it may be (re)used independent from the information stored in the learning environments. A default view on the course material is usually created by the instructor or the administrator.

Due to the individual style of learning it is absolutely necessary that such a default view is completely personalizeable. Not just the colors [6] and the style of the displayed text should be modifiable, but also layout must be exchangeable. This "personal workspace" must also be applicable to other courses. The view might also depend on the role the user currently has in the system (administrator, student, the person that prepares the course, instructur, ...).

Let us now consider some circumstances where it is necessary to change the view on the learning material. Take a fully featured multi-media course using different sources of information (integrating online and offline systems), different levels of detail (overviews but also information for specialists), etc.

transfer speed: Since we are working in a network environment, transfer speed is a big issue. The higher the available bandwidth, the higher the quality of e.g. the displayed images might be.

output device: Several output devices may be used for consuming a prepared course: Workstations, laptops, terminals, PDAs etc. As a simple terminal is not able to display images, why should the client on a terminal request them? So also the type of the output device must be considered when delivering information.

available sources: We mentioned the online- and offline systems and a combination of those, the hybrid systems using both technologies. For example a student using the course CD-ROM might be able to see a movie, whereas the students participating online (on a network with low bandwith) might only be able to view images instead.

consuming environment: When learning in a public environment (bus, train, ...), the user might want to disable e.g. the sound output, so this data does not need to be sent across the network.

level of knowledge: Depending on their actual knowledge of the learning material the users might not need every basic explanation on the topic of the course.

role: There might not just be an instructor and some students but also separate administrators or other roles. An instructor creates course material, but surly is not involved in administrating user accounts. The student may also be in the role of a candidate for an exam and may fill out an examination in a limited amount of time.

desired level of detail: The system should also consider the level of the desired detail set by the user. It must be possible to switch between different levels in one learning session.

personal preferences: Some users prefer different fonts, colors, screen-layouts, etc.

5 Conclusion

The *Adaptive Learning Environment Framework* presented in this paper is a powerful tool to add a surplus value to existing learning systems and to reduce the burden on students to get used to new interfaces to the material with every virtual class they enroll. Additional features relieve all participating persons, including the tasks of administration and preparation of courses.

It is obvious that the integration of different learning systems may be a very complex task, but the modular structure of Dinopolis including the internal document model is a powerful tool the learning community will not want to miss in the near future.

References

[1] Dallermassl Ch.: "Aspects of Integration of Heterogenous Server Systems in Intranets - The Java Approach", IICM (Institute for Information Processing and Computer Supported new Media) 1999

[2] Dinoopolis: "Homepage of the Dinopolis Open Source Project", available online at `http://www.dinopolis.org`

[3] Gamma E., Helm R., Johnson R., Vlissides J.: "Design Patterns: Elements of Reusable Object Oriented Software", Addison Wesley, 1995.

[4] Gentle-WBT: "Homepage of Gentle-WBT: Web Based Training", available online at `http://wbt.iicm.edu`.

[5] Hedberg, John G., Harper, B., & Brown, C.: "Reducing Cognitive Load in Multimedia Navigation." Australian Journal of Educational Technology 9, no. 2 (1993): 157-81.

[6] Kienegger-Domik G.: "Intelligent Visualization Systems in Educational Environments", Proceedings of ED- MEDIA'95, Graz, Austria, pp.17-22, June 1995

[7] McFarland, Ronald D.: "Ten Design Points for the Human Interface to Instructional Multimedia.", T.H.E. Journal 22, no. 7 (February 1995): 67-69.

An Educational System that can Visualize Behavior of Programs on the Domain World

Taketoshi Ando*, Yoshinobu Kawada*(†), Ryoji Itoh*,
Tatsuhiro Konishi*, Yukihiro Itoh*
* *Department of Computer Science, Shizuoka University.*
Johoku3-5-1, Hamamatsu, Shizuoka, 432-8011, Japan.
Tel: +81 - 53 - 478 – 1474• • Fax: +81 - 53 - 478 – 1499
E-mail: s5005@cs.inf.shizuoka.ac.jp

In this paper, we discuss extension of our educational system that gives domain-oriented-explanations of programs. A programmer solves problems on a world where elements necessary to describe the problems and the solving processes of them (objects in the problems, relation among the objects, and so on) are represented. We call such a world 'the domain world' of the problem. Our system has a domain world model and simulates a target program on the model, to understand behavior of the program. By analyzing the result of understanding, it generates an explanation. Outputs of our original system are only verbal explanations. However, when the system explains by using only sentences, some learners cannot get a concrete image of behavior of the program. Therefore, we are trying to add a facility of generating explanations by using animations (visual explanations) to the system. Our extended system can generate both visual and verbal explanations (bimodal-explanations) in various abstraction levels. We discuss the method of generating bimodal-explanations from the result of simulation.

Keywords: **Intelligent Tutoring System, Programming Education, Algorithm Animation System, Bimodal-explanation**

1 Introduction

The purpose of our research is to construct an educational system that helps novice programming learners by explaining domain-oriented-functions of programs. We take Pascal as our target programming language.

Programming is generally carried out in the following process.

Step1. A programmer understands a problem that must be solved.
Step2. He considers the solving process of the problem on a world where the problem is present. We call such a world 'the domain world' of the problem. For example, when he considers a solving process of sorting, he imagines a world in which he pays attention to numerical order such as greater and lesser (we call this world the world of greater and lesser).
Step3. He implements the algorithm: selects data structures suitable to represent the domain world and translates the algorithm into a programming language.

Usually, relatively simple problems are set in novice class of programming. So it is rare that learners fail in the step 1. But they tend to confuse because they cannot distinguish between step 2 and 3. So many novice programmers cannot find whether the causes of bugs are hidden in the algorithm or in their implementation. On the basis of this idea, we proposed an educational system explaining programs using vocabularies on a domain world[2][5][6][7]. Difference of our system from existing educational systems of programming [1][8] is that the purpose of our system isn't pointing out bugs in learner's programs, but rather helping learners find bugs by themselves. Our system helps learners in the following way:

- To help learners to understand sample programs given by a teacher by explaining them.
- To help learners to find and fix bugs in their own programs by explaining the faulty behavior of them.

Our previous system outputs sentences using vocabularies on a domain world as the explanation. However, when the system explains by using only sentences, some learners cannot get a concrete image of behavior of the

---

(†) *Presently with System Integration Group, VICTOKAI, LTD.*

program. If animations of the behavior of programs are shown with the sentences, learners can easily understand their algorithms. Therefore, we realize the ability to generate animations (visual explanations) that show behaviors of the target programs. In this paper, we discuss the way to generate visual explanations for programs in the domain world of greater and lesser.

Existing algorithm animation systems can be classified into two types: The first one is a system such as courseware editors embody particular commands to target programs in order to generate visual explanations, like Zeus[3] and TANGO[9] system. So, this type of systems can generate visual explanations of high quality by using concrete objects on the domain world. For example, a length of bar is used to concrete values of variables on the visual explanation of XTANGO system. The second type of system doesn't need embodying particular command to target programs, like UWPI[4] and tracers. However, this type of systems cannot generate any visual explanation using concrete objects on the domain world. They can only generate visual explanations showing structure of data and changes of contents of variables. Our system can generate a visual explanation using concrete objects on the domain world without embodying any special commands to programs. It generates visual explanations on the basis of the result of "simulation based program understanding[5]". So it can accept buggy programs and generate visual explanations of buggy behaviors of the programs. Moreover, it can also generate verbal explanations on the basis of the result of program understanding.

In the next section, we illustrate an overview of our previous system. In section 3, we point out some functions necessary to generate an effective explanation by using both verbal explanation and visual explanation (a bimodal-explanation). In section 4, we describe the method of constructing the bimodal-explanation system. Then, we show examples of bimodal-explanations by our system.

2 Our Previous Work

2.1 Overview of our previous system

Our previous system is composed of the static analyzer, the simulation based analytical unit and the explanation unit (Figure 1). In this paper, we omit detail of the system (For further details, please see our previous papers[2][5][6][7]). The static analyzer parses target programs and analyzes information necessary for the simulation such as data flow. The simulator simulates target programs, and the observer observes the world model while simulation, and recognizes some important characteristics of data or patterns of structured data. The explanation generator generates verbal explanations of target programs.

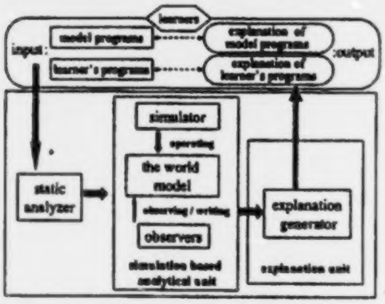

learners
input:
model programs
learner's programs
explanation of model programs
explanation of learner's programs
:output
simulator
operating
the world model
observing / writing
observers
static analyzer
explanation generator
simulation based analytical unit
explanation unit

Figure 1: Configuration of our system

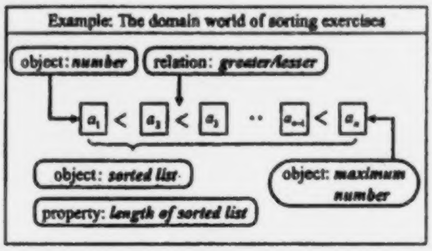

Example: The domain world of sorting exercises
object: number
relation: greater/lesser
a_1 < a_2 < a_3 ·· a_{n-1} < a_n
object: sorted list
object: maximum number
property: length of sorted list

Figure 2: An example of domain world

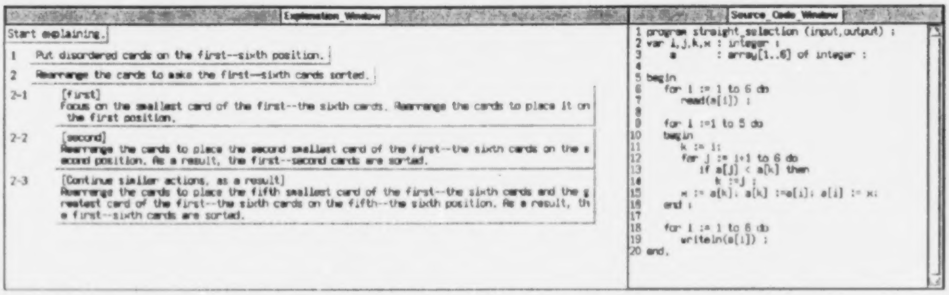
2.2 Domain world models

We examine programming exercises and classify them into 15 types. We prepare domain world models designed for each type of exercises [2].

A domain world model consists of four types of elements called 'object', 'property', 'relation among objects' and 'change'. For example, Figure 2 shows the domain world model of greater and lesser as an example. In order to recognize specified characteristic or patterns in the domain world, our system has daemon units called "observer" which are burnt when they are observed. In the Figure 2, the object *'maximum number'*, *'sorted list'* and property *'length of sorted list'* are recognized by observers. There are some cases that some observers take outputs of the other observers as their inputs. Then the outputs of observers make hierarchy. When a result of observation is output on the basis of a result of another observer, the former has larger grain-size than the latter and implies the fact corresponding to the latter.

2.3 Generation of a verbal explanation

The explanation unit generates verbal explanations of the target program by using results of simulation and outputs of observers. The results of observations have a hierarchical structure, as mentioned above. The system generates a hierarchical verbal explanation by using the hierarchical structure (it also uses syntactical structures of programs). In other words, the system notices the largest grain-sized result of the observation firstly, in order to generate the verbal explanation. Secondly, if learners request the detailed verbal explanations, the system generates the explanation using results of observation having smaller grain size. Figure 3 shows the example of verbal explanations generated by our system. It illustrates the verbal explanation of behavior of a sorting program on the domain world of greater and lesser. The indentation in the figure means that *behavior 1* and *behavior 2* are executed sequentially and that *behavior 2* is equivalent to the sequence of *behavior 2-1, behavior 2-2,* and *behavior 2-3*. Each Behavior is implemented by a single statement or a sequence of statements. When a verbal explanation for a behavior implemented by a sequence of statements is clicked, more detailed verbal explanations showing the way to implement the behavior are displayed.



Start explaining.

1 Put disordered cards on the first--sixth position.

2 Rearrange the cards to make the first--sixth cards sorted.

2-1 [first]
Focus on the smallest card of the first--the sixth cards. Rearrange the cards to place it on the first position.

2-2 [second]
Rearrange the cards to place the second smallest card of the first--the sixth cards on the second position. As a result, the first--second cards are sorted.

2-3 [Continue similar actions, as a result]
Rearrange the cards to place the fifth smallest card of the first--the sixth cards and the greatest card of the first--the sixth cards on the fifth--the sixth position. As a result, the first--sixth cards are sorted.

```
program straight_selection (input,output) ;
var i,j,k,x : integer ;
    a       : array[1..6] of integer ;

begin
   for i := 1 to 6 do
      read(a[i]) ;

   for i :=1 to 5 do
   begin
      k := i;
      for j := i+1 to 6 do
         if a[j] < a[k] then
            k :=j ;
      x := a[k]; a[k] :=a[i]; a[i] := x;
   end ;

   for i := 1 to 6 do
      writeln(a[i]) ;
end.
```

Figure 3: An example of the verbal explanation of a target program

Next, we show procedures for generating the hierarchical verbal explanation like Figure 3. An input is a result of simulation of some statements (For further details, please see our previous paper[7]).

(1). The case that a certain behavior is implemented by a sequence of statements.
The system observes differences of the states of the domain world model before and after execution of the sequence of statements. According to these differences, the explanation unit selects a template and generates verbal explanations for the statement. Now we show an example of a template.

- The case that extension of the sorted list is observed.
  The differences are composed of the following three elements.
    Object(s) recognized at the state before execution of some statements: a sorted list
    Object(s) recognized after execution of the statements : an extended sorted list
    Recognized changes of states of objects : an extension of the sorted list

A template for the extension of the length of a sorted list is applied (Please see Figure 3).
Template: "Rearrange the [Type of added object] to place [An added object] on [The position of the insertion] position. As a result, [A sorted list at the after state] [Type of inserted objects] are sorted."
"[]" means a procedure which generates a certain pattern of string.
[An added object]
: A procedure that generates a noun phrase expressing the new object added to the sorted list.
[A sorted list at the after state]
: A procedure that generates a noun phrase expressing the range of the sorted list at the after states.

(2). The case that a certain behavior is implemented by a single statement.
The explanation unit calls each procedure corresponding to types of the statement. The procedures are defined for each structure of the program like sequential structures, selective structures, iterative structures, an assignment statement, a statement for input, and a statement for output. Same as the case (1), templates are prepared for each structure of the program. For example, we show a template of 'if' statement.
Template: "if [explanations of the conditional clause], [explanations of the 'then' clause]
(otherwise [explanations of the 'else' clause]) "
[explanations of the conditional clause]
: The procedure that explains the conditional statement of 'if' statement.
[explanations of the 'then' clause]
: Apply the procedure for generating the verbal explanation to the clause recursively.
[explanations of the 'else' clause]

BEST COPY AVAILABLE

: Apply the procedure for generating the verbal explanation to the clause recursively.

Thus, the system can generate hierarchical verbal explanations. When a verbal explanation generated by the procedure (1) is shown and a learner requests more detailed explanation, the system tries to apply the procedure (1) recursively to make such an explanation. If it cannot generate any explanation, it applies the procedure (2).

3 Functions necessary to generate an effective bimodal-explanation

In order to construct a system generating effective visual explanations, we have to consider what visual explanation is effective for learners to understand an algorithm or behavior of a target program. By designing mock up visual explanations repeatedly, we find that the effective visual explanation has following three facilities.

(1) The facility to generate visual explanations with various grain-sizes.
When learners learn programming by using a system explaining behaviors of programs, they need various grain-sized explanations. For example, when a learner wants to grasp algorithm roughly, a large grain-sized explanation would be effective. On the other hand, when he wants to understand a precise method of implementation, smaller grain-sized explanations are effective. Moreover, when he wants to diagnose his own program at a glance, he needs the largest grain-sized explanation. When he wants to find buggy codes, he needs smaller ones. In order to generate such various grain-sized visual explanations, the system should be able to :

- regard a sequence of statements as a blackbox and generate a visual explanation showing its function.
- generate a visual explanation showing a function of each statement sequentially.

(2) The facility to explain a function of a program by using both animations and verbal texts.
If a system shows only visual explanations, learners sometimes cannot understand behavior of target programs clearly, because such learners cannot understand what phenomena are essential. Thus, it is necessary for our system to have the facility to generate verbal explanations showing a major phenomenon of each step of visual explanations. Thus our system should have a facility of generating combination of verbal explanations and visual ones (bimodal-explanations).

(3) The facility to generate explanations on the total effect of a sequence of statements.
Generally, a task is achieved by a sequence of statements, and each sub-task is achieved by each sub-sequence of the statements. When the system shows a sequence of explanations each of which has a certain grain-size corresponding to a sub-task, a learner sometimes cannot find the fact that the task has been achieved. In order to prevent learners from such misunderstanding, the system should show them a verbal explanation remarking the fact.

4 Methods to realize the functions to generate bimodal-explanations

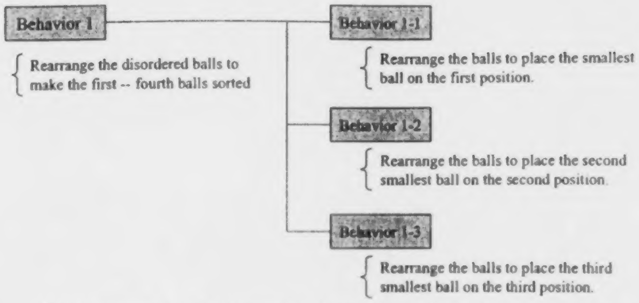
4.1 Basic ideas

(1) The method of generating visual explanations on various grain-size.
As we describe in section 2, our system can generate hierarchical verbal explanations. In other words, it can understand behavior of a target program on various grain-size. And the system holds the result of understanding as hierarchical data. Therefore we can realize a system generating visual explanations on various grain-size, by developing a method to generate a visual explanation from a result of understanding.

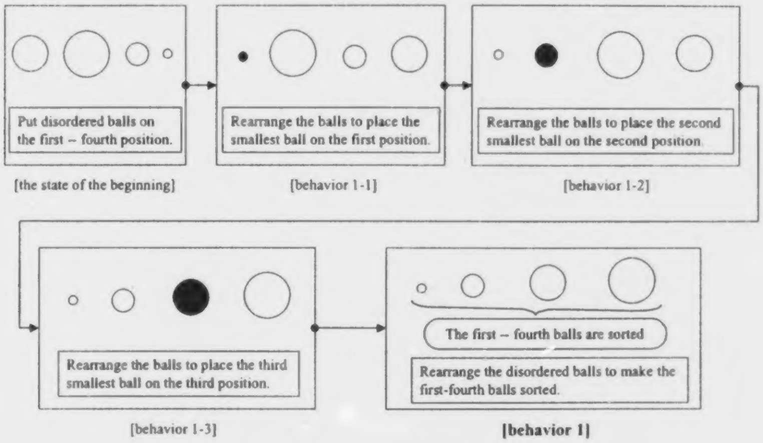
(2) The method of generating combination of verbal explanations and visual explanations.
Our program understanding mechanism can recognize the major phenomena in the domain world. And we have already developed a method to generate verbal explanations from the result of program understanding. Thus, if the system can generate a visual explanation from the result of it by the method (1), it becomes to be able to generate both visual explanations and verbal explanations remarking major phenomena from common data.

(3) The method of generating explanations on the total effect of a sequence of statements.
By generating an explanation remarking that a task is achieved just after explanations of sub-tasks are finished, the system can generate explanations on the total effect of the task. The explanations of the task and the sub-task can also be generated by the method (1) and (2). For example in Figure 4, just after the explanation corresponding to the behavior 1-3 is finished, the system generates the explanation corresponding to the behavior 1 as the explanation of the total effect. As a result, the explanation shown in Figure 4 is generated.

In consequence, if we can realize the method (1), the method (2) and (3) can also be realized. Therefore we discuss the detail of the method (1) in the next section.


Behavior 1
Rearrange the disordered balls to make the first -- fourth balls sorted
Behavior 1-1
Rearrange the balls to place the smallest ball on the first position.
Behavior 1-2
Rearrange the balls to place the second smallest ball on the second position.
Behavior 1-3
Rearrange the balls to place the third smallest ball on the third position.

(a): An example of a hierarchical structure of behaviors of a program


Put disordered balls on the first – fourth position.
[the state of the beginning]
Rearrange the balls to place the smallest ball on the first position.
[behavior 1-1]
Rearrange the balls to place the second smallest ball on the second position.
[behavior 1-2]
Rearrange the balls to place the third smallest ball on the third position.
[behavior 1-3]
The first – fourth balls are sorted
Rearrange the disordered balls to make the first-fourth balls sorted.
[behavior 1]

(b): Generated explanations

Figure 4: An example of a bimodal-explanation of a total effect

4.2 Generating visual explanations

The system visualizes behavior of the target program in various grain-size. The generated animations are shown with verbal explanations. The detail of our method to generate verbal explanation is seen in [6], so we omit it in this paper.

At first, the system starts explaining with the largest grain-size, then shows more detailed explanation on an action of which detail a learner wants to see.

The methods to draw a step of animation are classified into the following two types:
1) The method of visualization for a function implemented by a single statement.
2) The method of visualization for a function implemented by a sequence of statements.
The detailed process of 1) and 2) is discussed in 4.2.1 and 4.2.2 respectively.

4.2.1 How to generate a visual explanation of a function implemented by a single statement

In order to generate a visual explanation on a statement, we prepare specific procedures for each type of a statement. The statements of inputting, assignment, selection, and iteration have their individual procedures.

Procedures for inputting statement should be classified into several types in order to generate effective explanations. For example, the basic function of inputting statement "read (A);" must be "a datum is input to the variable A". However, showing only the basic function is not always a good explanation. If a meaningful datum has been stored in the variable "A" before inputting, the system should also explain that the datum is deleted by the inputting. Therefore, the procedures for inputting statement are classified according to some conditions on the role of the statement in the target program and the domain world: for example, the condition whether the datum stored in the destination variable of inputting has been referred before the input sentence or not (if it has been referred, it must be meaningful).

An Empirical Study of the Design and Use of a Multimedia Composition-Making System for Children

Kaori Iwashita*, Shinya Kawakami and Bipin Indurkhya**
Department of Computer Science, Tokyo University of Agriculture and Technology,
Koganei City, Tokyo, Japan, tel: 81-42-364-3311
**e-mail: kaorii@ma.neweb.ne.jp, **e-mail: kawakami@fairy.ei.tuat.ac.jp*

In this paper, we describe our experiences in designing and using a multimedia composition-making system for children. The system allows children to make compositions using pictures, sounds and text. Moreover, it also allows pictures in the composition to be animated. We experimented with children using this system in three different settings. In the first setting, no topic was assigned to the children. In the second and third experiments, children were given a topic (different for each experiment) for composition related to their activity. We present here the results of our experiments and comment on how the constraints imposed by the topic affect children's expressive abilities.

Keywords: **animation, children's expressive abilities, constraints and creativity, multimedia composition.**

1 Introduction

In recent years, many researchers have studied multimedia techniques and have incorporated them into various educational systems. For example, Silva [5] described a multimedia soundscape system, "They Are Catching Sounds in the Park!", for environmental education. In this system, children search for sounds by clicking anywhere in the picture. When they click an appropriate object, its associated sound and information are presented to the children. Brna [1] proposed a system for composing and writing stories via cartoons. Harviainen [2] presented a co-authoring system in which many users work together to compose a story. Ishii [3] and Kawakami [4] have developed other systems for making stories with multimedia. All this research demonstrates that multimedia has much potential for stimulating the ability of children to express themselves. In particular, we find that children can express their creative and imagined ideas much better with pictures and words than with words alone. Moreover, if we add an ability to attach sounds to pictures, and allow pictures to be animated, then this expressive power increases considerably.

Motivated by these factors, we have developed a system to help children write multimedia compositions, and have tested it with children in three different settings. In this paper, we describe our system and report on our experiences with children using the system.

2 Prototype of the System

We developed a prototype of a multimedia composition-making system. Using our prototype,

- Children can express their thoughts and ideas via pictures, sounds, text, and animation sequences. In our system, children must first choose a background scene, in which they can then insert picture objects, sounds, and text.
- Except for the background, children can attach sounds and text to picture objects, and can animate them to make a multimedia composition.

This system has two modes: a 'Set up' mode for the teacher or supervisor to allow them to determine which background scenes, picture objects, sounds, etc. are made available to the children for writing a composition, and a ' User' mode for children to write compositions.

The ' Set up' mode has the following two functions:
1) Select situation: Set the context or theme for the composition.
2) Edit situation: Set the categories of background scenes, picture objects and sounds corresponding to a theme.

The ' User' mode has the following seven functions:
1) Select background scenes.
2) Select picture objects.
3) Select sounds.
4) Write text.
5) Animate composition.
6) Save composition.
7) Load composition.

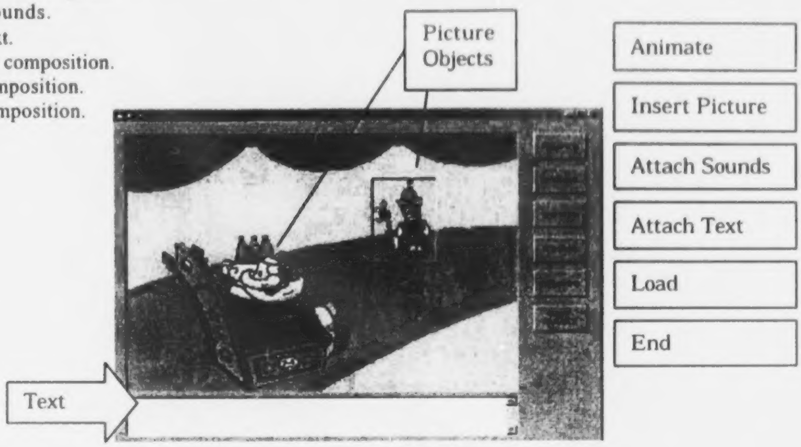


Figure 1. An example of the main window

By double clicking on a picture in the main window, the sound attached to that object (if any) can be heard. Also, when the picture of an object is selected in the main window, the text attached to it is displayed in the text box.

The animation module has five functions: show picture, hide picture, output sound, show text, and move picture.

To replay animation, children click the ' start' button in the animation window. When the button is clicked, the system starts the animation sequence as previously specified. It replays each action one by one, but it pauses when the action is ' show text' . To continue from there, the user needs to click the ' start' button again.

3 Experiments with the system

We did three different experiments in which children used our system. In each experiment, the setting and the tasks given to children were different, as described below.

3.1 Experiment I

In this experiment, we studied a constraint-free use of the composition system. The children were not given any specific topic of composition and they could use the system any way they like to create any composition freely. We prepared 54 background scenes, 185 pictures and 68 sounds. Because no topic was given, children chose a variety of themes.

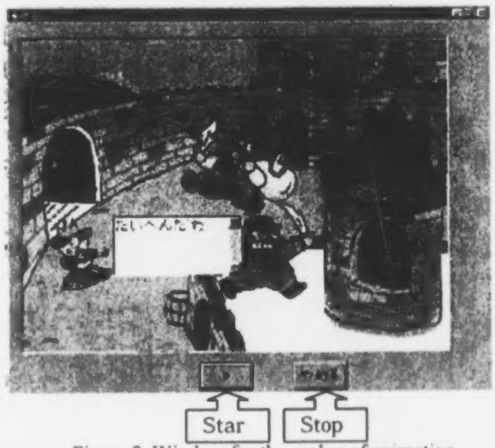


Figure 2. Window for the replay of animation

BEST COPY AVAILABLE

3.2 Experiment II

In this experiment, we introduced a constraint by giving a topic of composition to the children, and analyzed the generated compositions. The experiment was done at an activity center for children. At this center, children of each grade come periodically, and play or make some handicraft. One of the handicraft projects for third-grade children was making kites. So, the following week, we asked the children to make a composition about kite making. For the experiment, we prepared a version of the system with six backgrounds scenes of craft rooms. Three of these were scenes with kites in them, and the others were scenes with only a room and tables without kites. We also prepared 68 pictures and 35 sounds appropriate for kite-making activity.

3.3 Experiment III

In this experiment, we introduced a tighter constraint by giving a more specific topic of composition to children, and studied its effect. We asked the children to make a composition for the story "The coward king and robber" (original title in Japanese). The original story is written in Japanese. At the same activity center for children used in Experiment II, the children made an extended version of this story, made a picture book to illustrate various scenes in the story, and then told the story using these pictures at their Christmas party. The week following the party, we asked the children to make a composition for this story using our system.

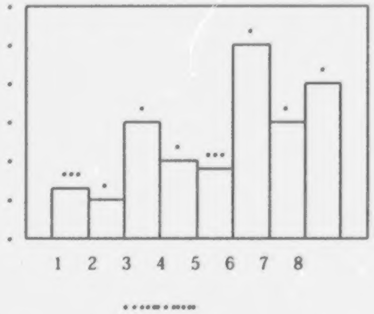


Figure 3. Differences between the composition picture and the original picture from the picture book (for each background scene)

For this experiment, we prepared a version of the system with eight background picture scenes related to the story. We also prepared 66 picture objects and 33 sounds appropriate to the story.

In this experiment, we were interested in analyzing the differences between compositions made using our system and the corresponding pictures in the picture book for this story that the children had made earlier. We used the following method for computing the difference. The picture objects were grouped into ten categories, and the difference between two pictures (with the same background scene) was calculated as follows:

> For each picture object category: if there is an object of that category in both the pictures, we say that the difference between the two pictures with respect to that category is zero. If one picture has an object from that category, and the other has none, we say that the difference with respect to that category is one. The difference between two pictures is the sum of the differences over all ten categories.

Figure 3 shows the result of applying this procedure.

We see that the differences for the climactic scenes (scenes 6• 7) are higher than the other scenes.

3.4 Discussion

An analysis of the compositions produced in the three experiments is shown in Tables 1, 2 and 3. Table 1 shows the average number of compositions produced by a participant in each experiment. We see from it that the children were most expressive when the topic was most constrained (Experiment III).

Table 1. Number of compositions per participant

	Experiment I	Experiment II	Experiment III
Number of participants	11	7	5
Number of compositions per participant	1.3	1.4	4.6

Table 2 shows a more detailed analysis of compositions with respect to how multimedia features of the system were used.

BEST COPY AVAILABLE

Table 2. Number of multimedia features per composition

Multimedia feature	Experiment I	Experiment II	Experiment III
Picture objects	3.6	11.2	6.9
Sound attachments	1.1	2.4	1.6
Text attachments	-	1.1	1.6
Animation	1.0	2.8	9.9

Here we see that picture and sound attachments are used most in Experiment II. This may be because in this setting children were describing a situation using generally one page (screen). For this, they used many objects and sound attachments to provide information about the depicted situation. We also see that text attachments and replay actions were used most in Experiment III. It might be because in this setting they were describing a story, for which text attachment is a useful way to express characters' utterances, and animation is useful to express characters' movement. We also would like to point out that in Experiment III there were fewer picture objects and sound attachments per composition. This is because to show the flow of events in the story, children made many compositions (Table 1).

Table 3. Analysis of animation operations per composition (in percent)

Animation operation	Experiment I	Experiment II	Experiment III
Show picture	0	0	53
Hide picture	0	0	4
Output Sound	13	0	10
Show text	-	0	15
Move picture	87	100	18

In this table we see that in Experiments I and II, mostly the ' move picture' operation was used. In composing a story, however (Experiment III), the 'show picture' was used most. We also found that the ' hide picture' operation was rarely used.

4 Conclusions

From our experiments, we see that the multimedia features of the composition-making system are most useful in illustrating a story or a narrative. Sound and text attachments and animation operations can be very helpful in expressing movement of characters and the progression of events in a narrative. We also found that many children are most expressive when they are given a focus of composition.

From these results, we propose that a system such as ours can be used in the classroom for children to make compositions about field trips and class excursions. For each trip or excursion, the teacher can set up the system appropriately by choosing relevant picture and sound libraries before children use the system. In this way, we feel that our system can provide a step forward from Silva [5]. Children are more actively involved in making compositions with our system than in exploring with ' They are catching sounds in the park!"

Acknowledgements

We would like to thank all the children who participated in the experiments, and the staff at the children's center for their cooperation. We thank Professor Yoshiyuki Kotani, and members of the Kotani-lab for their help and cooperation during this research. Some pictures used in the background of Experiment III were taken from [6].

References

[1] Brna, P., Martins A., & Cooper B., "My First Story: Support for Learning to Write Stories", Advanced Research in Computers and Communications in Education, pp. 335-341, 1999.

[2] Harviainen, T., Hassinen, M., Kommers P., & Sutinen, E., "Woven Stories: Co-authoring over the Internet", Advanced Research in Computers and Communications in Education, pp. 285-292, 1999.

[3] Ishii, Y., Indurkhya, B., Inui, N., Nose, T., Kotani, Y., & Nishimura., H., "A System Based on Rodari's ' Estrangement' Principle to Promote Creative Writing in Children", ED-MEDIA & ED-TELECOM 98,

Vol. 2, 1685-1686, 1998.

[4] Kawakami, S., Indurkhya B., Inui, N., & Kotani, Y., "A Story Making System for children", Advanced Research in Computers and Communications in Education, 359-360 1999.

[5] Silva, M.J., Silva, J.P., & Hipolito, J., ""They Are Catching Sounds in the Park!" Exploring Multimedia Soundscapes in Environmental Education", ED-MEDIA & ED-TELECOM 98, Vol. 2, 1274-1279, 1998.

[6] Yamawaki, K., & Miyamoto, E., "Yowamushi o-sama to to-zoku", Kyoiku Gageki, 1989.

An Implementation of Campus Distance Learning System Using Multicast

Jiang Dongxing, Tao Chaoquan, Fang Guowei, Shen Peihua
Computer and Information Management Center of Tsinghua University
Qinghuayuan No.1, Beijing 100084, P.R.China
Tel: +86-10-62772150, Fax: +86-10-62782728, E-mail: jdx@cic.tsinghua.edu.cn

A problem common to many universities is that thousands of students want to take some certain courses but only a few can actually take them owing to the shortage of teachers. The Campus Distance Learning System is an important way to solve the problem. This paper starts with an examination of some existing solutions, and then introduces the primary-secondary model multimedia network teaching system designed by researchers in the Computer & Information Management Center of Tsinghua University. The system is composed of three parts: the primary classroom system, the secondary classroom system, and the courseware management system. It fulfills real-time interactive teaching and learning, and multipoint communication, and at the same time records the teaching materials as courseware. The paper focuses on the constituents, structure and characteristics of the system, and expounds in detail the implement technology based on multicast. In the end, the paper points out some problems calling for further consideration.

Keywords: **Network distance learning; primary-secondary model; multicast**

1 Introduction

A problem common to many universities is that thousands of students want to take some certain courses but only a few can actually take them owing to the shortage of teachers. The traditional resolution was videoing and then broadcasting through CATV. This used to play an important part in television education, but it can not support the interaction between the teacher and the students, and the information that is limited by TV is not sufficient for lectures. With the network becoming more and more popular, network education instead of CATV is being received by more and more people. Many companies and universities have developed different network distance learning systems, the following are several famous systems.

Remote Education System of VTEL: This system is an application of the VTEL videoconference system in education. It adopts a complete set of software and hardware developed by VTEL and can implement mulitpoint bi-directional interactive network education.

IP/TV of CISCO: IP/TV is software developed by Cisco company, supporting video on demand and video broadcast. It adopts the client/server model and is mainly used for transferring high quality video, audio and data via computer networks. The system supports three ways of video transferring: live, on-demand and scheduled.

Multimedia Distance Education System of SATCOM: This system includes a program courseware generation system and a courseware on demand system.

2 Primary-Secondary Model Multimedia Network Teaching System

2.1 System Constituents

The primary-secondary model multimedia network teaching system is composed of three parts: the primary classroom system, the secondary classroom system and the courseware management system (See figure 1).

The primary classroom is where the teacher stays. In this classroom, the video and slide of the lecture are recorded synchronously. The video and slide information is broadcast live through multicast, and at the same time the information is stored in the courseware library for asynchronous use.

The secondary classroom is the classroom without the teacher, maybe a remote classroom. Students in this classroom can join the lecture by registering and playing the composite stream courseware synchronously with the primary classroom.

The courseware management system provides the directory service, user register management, asynchronous courseware on demand, and other management functions of the coureware library.

The free terminal can join the lecture from anywhere of the campus network through registering. It can also play courseware on demand from the courseware management system.

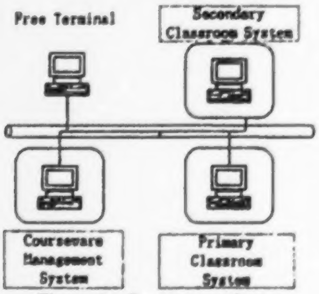

Free Terminal
Secondary Classroom System
Courseware Management System
Primary Classroom System

Figure 1 System constituents

2.2 System Structure

Figure 2 shows the structure of the primary-secondary model multimedia network teaching system. Its subsystems are as follows:

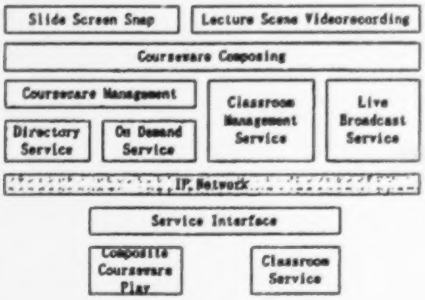

Slide Screen Snap
Lecture Scene Videorecording
Courseware Composing
Courseware Management
Directory Service
On Demand Service
Classroom Management Service
Live Broadcast Service
IP Network
Service Interface
Composite Courseware Play
Classroom Service

Figure 2 System structure

- Courseware synthesizing: The courseware synthesizing is the kernel subsystem of the primary classroom system. In this procedure, both the basic materials of the courseware – lecture scene videorecording and slide screen snaps are compressed into the composite courseware with synchronous timestamp. Afterwards, the courseware is stored into disks and multicast at the same time by the system.
- Lecture management service: The lecture management service is anther important subsystem of the primary classroom system, its main functions being registering new courseware in the courseware management system, requesting for the multicast address, configuring the multicast scope and lecture management.
- Directory Service: This is the kernel function of the lecture management system. It provides lectures and courseware lists and user management.
- On-demand Service and Live Broadcast Service: On-demand service is an asynchronous courseware service provided by the courseware management system while live broadcast service is a synchronous

BEST COPY AVAILABLE

service provided by the lecture management system. Both of them provide composite stream courseware to the user, the former using unicast and the latter using multicast.

- Lecture Service: This is an interactive supporting system provided by the secondary classroom system. With it students in the secondary classroom can participate in the discussion. The means of interaction may be keyboard typing, and speaking with a microphone.

2.3 System Characteristics

The main characteristics of the primary-secondary model multimedia network teaching system are the following:

1) It uses two streams to play the teacher's videorecording and slide screen snaps, and the quality of the slide screen snaps is the same as that of the slides in the primary classroom.
2) The lecture scene is kept in the archives in real time, and can be replayed at any time.
3) The teacher can discuss with students in remote classrooms through videoconference, and they can write on the same electronic white board.
4) The audience can have interlocution with the lecturer by text typing.

3 Implementing the System with Multicast

3.1 The Multicast Technology

By keeping routers informed about multicast hosts, multicast datagrams can traverse an internetwork and reach many hosts simultaneously. The ability to traverse an internetwork and reach an unlimited number of "member" hosts simultaneously without affecting others adversely is the linchpin of multicast. A Class D IP address in the range from 224.0.0.0 to 239.255.255.255 is a "multicast address." Each is also known as a "host group address," since datagrams with a multicast destination address can be received by all hosts that have joined the group that an address represents. Figure 3 shows the datagrams spreading abroad.

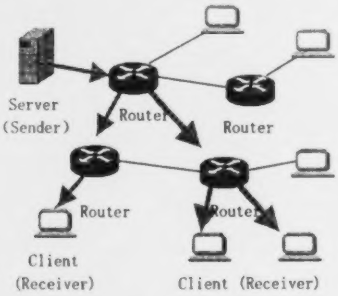


Figure 3 Datagrams sent only to hosts in a group

The mechanisms incorporated into WinSock 2 for utilizing multicast capabilities can be summarized as follows:

- Three attribute bits in the WSAPROTOCOL_INFO struct, which are used by **WSAEnumProtocols()** to discover whether multicast communications are supported for a given protocol;
- Four flags defined for the *dwFlags* parameter of **WSASocket()**;
- One function, **WSAJoinLeaf()**, for adding leaf nodes into a multicast session;
- Two **WSAIoctl()** command codes for controlling multicast loopback and the scope of multicast transmissions (SIO_MULTICAST_SCOPE and SIO_MULTIPOINT _LOOPBACK).

We can benefit from using multicast to implement network teaching system, which can be described as the following:

1) Because the member of a multicast group is dynamic, and no authority is requested, the terminal can join or quit a group at any time;
2) All hosts belonging to a multicast group have a clear physics network topology;
3) All users in one subnetwork that join the same multicast group share the same stream over network, and this can greatly lighten the network load.

3.2 System Implementation

In the practical system, we adopt the combinative way of multicast and unicast: using multicast to broadcast information from the primary classroom, and using unicast to implement the interaction between the primary classroom and the secondary classroom. Figure 4 shows the structure of the practical system in detail.

The primary classroom system is composed of a server, a teacher's PC, a video recorder, two overhead projectors and an electronic white board. The teacher's PC is used to play slide of the lecture, and it projects the slide to the electronic white board. If the teacher writes something on the electronic white board, the teacher's PC will capture the written information and combine it with the slide. At the same time, the teacher's PC compresses the slide/written information and sends it to the server. The server takes charge recording the video/audio information, receiving the slide/written information from the teacher's PC, and broadcasting the information with multicast. At the same time, the server stores all information into special type file, which is the composite courseware.

The full function secondary classroom is made up of a server, a video recorder, two overhead projectors and an electronic white board. The server receives the video/audio and slide/written information from the primary classroom, and projects the video information on the white wall, the slide/written information on the electronic white board.

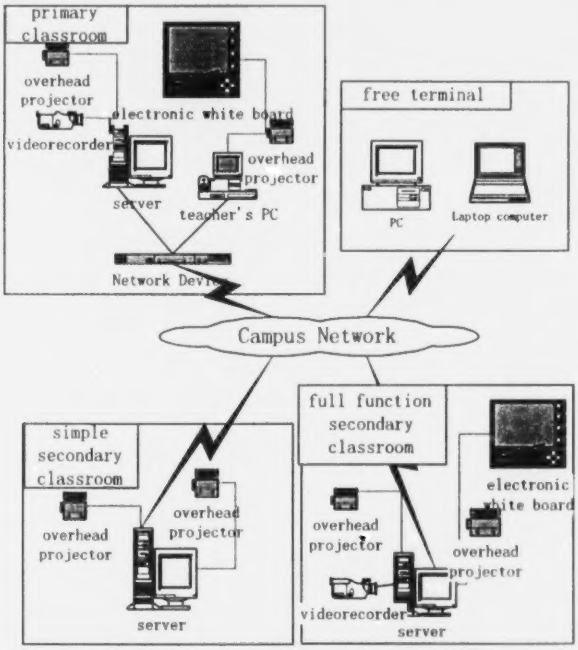

primary classroom
overhead projector
electronic white board
videorecorder
server
overhead projector
teacher's PC
Network Device
free terminal
PC
Laptop computer
Campus Network
simple secondary classroom
overhead projector
overhead projector
server
full function secondary classroom
electronic white board
overhead projector
overhead projector
videorecorder
server

Figure 4 The topology of the primary-secondary model multimedia network teaching system

The simple secondary classroom is made up of a server and two overhead projectors. The server receives the video/audio and slide/written information from the primary classroom, and projects the information on the white wall separately.

The free terminal may be any PC connected to the network. It receives the video/audio and slide/written information from the primary classroom and displays it in different windows.

The teacher in the primary classroom and the students in the secondary classroom can discuss with each other. This is implemented with unicast. During the discussion, the server in the secondary classroom records the information of the students and sends it to the server of the primary classroom. The server of the primary classroom receives this information and projects it on the white wall. If students in the secondary classroom write something on the electronic white board, the servers will transmit the written information, which will be shown on the electronic white board of the primary classroom at the same time.

4 Conclusions

In our experiments, we use lossless a compression algorithm and the slide screen snaps can be compressed to 1%~2%, it means that the slide screen snaps will take up 100~200Kbps bandwidth. In another side, the video information can be compressed into 128Kbps by MPEG-4 and all the information of this system can be fit in a 384Kbps channel. This system is available for long distance learning and of course for campus distance learning.

The primary-secondary model multimedia network teaching system has built a virtual network classroom system. It will play an important role in making better use of teaching resources and improving teaching efficiency.

References

[1] Jiang Dongxing, Lin Ehua, "Windows Sockets Network Programming", Tsinghua Univ. Press, (1999).

[2] Colin Perkins and Jon Crowcroft, "Real-Time Audio and Video Transmission of IEEE GLOBECOM'96 over the Internet", IEEE Communications Magazine, pp.30-33, (1997).

[3] Victor O. K. Li , Wanjun Liao, "Distributed Multimedia Systems", Proceedings of the IEEE, (1997).

[4] Song Yunxian, "The present and future of network education", Multimedia World, (1997).

Building the Multi-tier Architecture of Component-Oriented Multimedia CAI Systems on Internet

Min-Huei Lin , Ching-Fan Chen, Chen-Nan Chiang
Department of Information Management
Aletheia University
Tamsui, Taiwan, R.O.C
Email: lmh@email.au.edu.tw, cfchen@email.au.edu.tw

The popularity of WWW (World Wide Web) produces lots of new instructions or substitutive cases to build a new future, therefore educational units need to develop various computer-assisted instructions. To ensure good learning effect, the instructive strategy adopted by most CAI systems is to provide tremendous amount of multimedia data in order to attract the learner and a complete process of instruction is like the scenario of a presentation. The purpose of this thesis is to discuss how the multi-tier developing architecture can let the multimedia learning resources be used and shared in WWW from a view of organization's requirements, such that teachers, measuring researchers, and learning researchers can perform different tasks according to their own specialties independently. We also propose and implement a multimedia presentation system to let various authors with various identities author and present their presentation, i.e. CAI systems, conveniently and correctly. We compare the general hierarchy of a multimedia presentation system with the multi-tier architecture proposed by us, and we can know how the tasks are divided and assigned to corresponding professionals to accomplish the whole teaching materials through working cooperatively. It is possible to have a suggestion to develop CAI software for educational department.

Keywords: **Multimedia Presentation System, CAI System, Multi-tier**

1 Introduction

Although there exists many arguments, object-oriented is still spread out in 1990's and it seems to be a possible survival direction in software crisis. Besides this, we can use component oriented to build a set of CAI systems via existing papers that can be divided into several areas, e.g. research of interface, learning methods of computer assisted instructions, application of virtual reality, networking exam, virtual classrooms (including distance instruction), individual researcher objects, and etc. For example, the processes of mental model research emphasize the use of information of objects, so researchers just make the analysis components of mental model, the key point of this study is the component of mental model, not the scenario of teaching and the interface of designation. Another example, fuzzy theory should be used in the research of learning analysis, the key point is to provide learning analysis for content of exam, and it can make the analysis component purely. From the two examples, we can find the generation in proper components analysis, so all we have to do is making the component of its own domain. Each researcher only concerns its own theme without being concerned with the entire system, then can reuse the resources and get the complete experimental environment. This thesis constructs the developing architecture of CAI through component oriented and logical dividing of multi-tier structure, and emphasizes that the discussion of developing architecture is the beginning of the series of research.

2 Multimedia presentation system

BEST COPY AVAILABLE

2.1 General Hierarchy of Multimedia Presentation System

On Internet, the way to play multimedia objects is hypermedia shown in the Fig.1. To display such a scene on homepages, we can divide the designation into two layers, frame layer and resource object layer. The resource object layer stores all the multimedia objects participated in playing, the frame layer records the objects that compose each frame, the schedule of playback, the arrangement of objects on screen, and the events that may change the playing flow of inter-frames.

A multimedia resource may be a picture, a text description, a video, or other materials that can be used in a multimedia computer. A topic is a resource carrier that presents the resource to the addressee. A frame is a composite object that represents related issues that a presenter wants to illustrate. A frame may contain push buttons, one or more topics to be presented, and a number of knowledge. A message with optional parameters can be passed between two frames (or back to the same frame) to trigger a multimedia presentation action.

In the two layers, we make some definitions by referring the various links defined in [7].
An inheritance* (successor or precedence) *link: is a property inheritance between two frames and is used in the process of knowledge collection of an activated frame before the logical inference of the frame proceeds.
A usage link: is a link that represents a message passed between two frames.
An aggregation link: indicates that a frame is using a resource.
A resource association link between two resources: indicates that the two resources are correlated.
A frame association link between two frames: indicates that the two frames are correlated.

2.2 Models of Presentation systems

In 1983, James F. Allen advocated in ACM• There exist thirteen temporal relationships between two intervals, namely, before, meets, overlaps, during, starts, finishes and the other six inverse relations as well as equal. The thirteen corresponding temporal operators constructed from the Allen's interval-based temporal logic are depicted in Fig. 2.

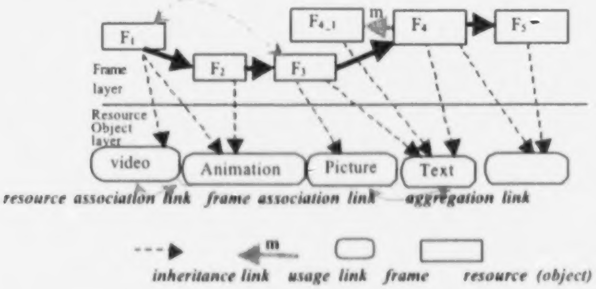
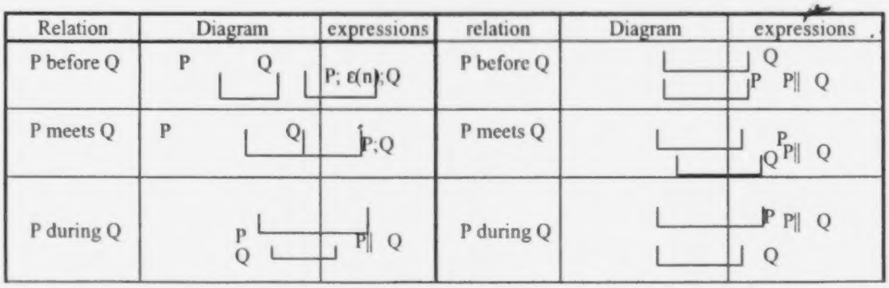


Fig.1 the way to play multimedia objects on Internet is hypermedia



Relation	Diagram	expressions	relation	Diagram	expressions
P before Q	P Q	P; ε(n);Q	P before Q		Q P P‖ Q
P meets Q	P Q	P;Q	P meets Q		P Q P‖ Q
P during Q	P Q	P‖ Q	P during Q		P P‖ Q Q

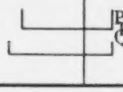
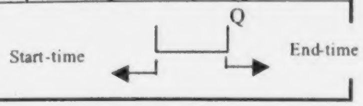
Relation	Diagram	expressions	relation	Diagram	expressions
P finishes Q		P || Q			

Fig. 2 Temporal Relationships (Allen 1983) and their corresponding temporal operators

2.3 Define the Playback of Multimedia Presentation

We define some notations used in our presentation system. The F_i denotes the frame in the frame layer. The O_i denotes the resource in resource layer. The $F_i.O_i$ denotes that the resource O_i is one component of the frame Fi. The m_i denotes a triggered message when users push a button, a hypertext or a hypermedia. The m_iF denotes that the frame F will be displayed after the message mi is triggered, and the m_eF denotes that the frame can be directly displayed not depend whether the message is triggered or not.

For example, a presentation displayed one frame by one frame can be described by the following expression $S = m_eF_1\ (m_iF_{21}+m_eF_{22})\ m_eF_3\ m_eF_4\ (m_iF_{51}+m_eF_{52})$. According to Fig.1, we know that the $F_i.O_i$ is an aggregation link, m_eF_1 is an inheritance link, and m_iF_1 is a usage link.

2.3.1 Define the Properties of scenario

A complete process of instruction is just like the scenario of a presentation, and can also be described by the expression $S = m_eF_1\ (m_iF_{21}+m_eF_{22})\ m_eF_3\ m_eF_4\ (m_iF_{51}+m_eF_{52})$.

2.3.2 Define the Properties of Objects

We denote a media object as O = (N, T, D, UM, OAL, PT), and describe the attributes of an object below:
O_i.N *(Name)*: is the identifier of the object.
O_i.T *(type)*: What multimedia device is used to carry out this resource (e.g. sound, video, text or picture).
O_i.D *(Duration)*: records the display time of the object.
O_i.UM *(Usage model)*: the situation about the usage of objects, such as the object is a background or a referent.
O_i.OAL *(object association link)*: describes the relationships between objects, and is specified like **O_i.FAL**={O_1(*association Keyword description*), O_2 (*association Keyword description*) ,..}, we use association keywords to describe the related relationships between O_i and O_1, the same as O_i and O_2.
O_i.PT *(Player Type)*: describes the way to play the object.

2.3.3 Define the Properties of Frames

A frame F_i is denoted as F_i =(N, O, FAL, L, P, UM), and the meanings of its attributes are listed below:
F_i.N*(Name)*: assign a unique name to a frame F_i.
F_i.O*(resource objects)*: the set of all the resource objects participated in the frame F_i, $O = \{O_i \mid O_i \in O\}$.O is the set of all objects stored in database.
F_i.FAL*(frame association link)*: $F_i.FAL = \{(C_i, F_k) \mid C_i \in \{$◈, $\odot\}, F_k \in F\}$.The relationships between F_i and F_j are divided into inclusive and exclusive relationships; we denote them by $\odot$ and ◈ respectively. The $F_i \odot F_j$ represents the two frames are inclusive, that is, whenever the F_i is displayed, the F_j must be displayed also. The F_i◈F_j represents the two frames are exclusive, that is, whenever the F_i has been displayed, the F_j can't be displayed. F is the set of all frames.
F_i.L *(Layout)*: the spatial arrangement of the objects of F_i for the presentation. For example, the (X_{j1}, Y_{j1}) and (X_{j2}, Y_{j2}) are the position on the screen arranged for O_j, $F_i.L = \{O_1\ (X_{11},Y_{11})(X_{12},Y_{12}),\ O_2\ (X_{21},Y_{21})(X_{22},Y_{22}), \ldots\}$.
F_i.P *(Presentation)*: the duration of playback of all objects in the F_i. We use the 13 temporal relations proposed by Allen and use $\varepsilon(n)$ to represent units of time. OP is the set of all operators used to describe the temporal relations between objects. P is a set composed of O_i OP O_j, $P=\{(O_i, op, O_j) \mid O_i, O_j \in O, op \in OP\}$, $OP = \{\ ;, \|, \|, \|, \|, \|, \varepsilon(n)\}$, $F_i.P=\{\cup P_i \mid P_i \in P\}$. For example, $F_i.P=\{(O_1 \| O_2),(O_1 \| O_3),(O_3\ ;\ O_4),..\}$.
F_i.UM*(Usage model)*: describes usage of frames, e.g. the frame is designed for teaching or for taking exams. For example, expression F_i.UM = exam means that the frame is an exam frame.

BEST COPY AVAILABLE

3Three-layer CAI architecture

3.1 Partition the CAI system into Components

The flow of instruction is from teaching course, taking examinations, speculating the advanced contents of instruction according to the result of examination, to achieve the goal of instruction. Generally, the teachers, educators or scholars take part in editing the CAI systems and the computer engineers are responsive for implementing the CAI systems, so they often spent lots of time on mutual communication. We analyze the CAI systems and partition the CAI systems into various components that are designed by various persons respectively, and these persons work together to achieve the whole function of the CAI systems. To partition the components clearly, we use the UML to describe the flow of CAI systems shown in the Fig.3, and we can know the following things:

- Step1 to step4 is for identifying the users.
- Step5 to step8 is for displaying the teaching of courses or questions of exams.
- Step9 to step11 is for analyzing after the exams are finished.
- Step12 to step14 is for designing the advanced courses after the fitting analysis is finished.
- Step15 is for exiting the CAI system.

In Fig.3, we can classify the partitioned components of CAI systems into four kinds listed below.

- The verification component for logging the usage of systems and maintaining the security of system. —is managed by system administrators or computer engineers.
- The course and exam component for instructing students in learning and taking exams. —is managed by teachers, educators or scholars.
- The fitting analysis component for the learning process of students. —is made by educators and scholars.
- The database component for storing the media objects and instruction materials. —is implemented by art designers or computer workers, and is managed by computer engineers.

3.2 Three-layer CAI architecture

From the CAI system described with UML shown in Fig.3, we can know that the course and exam component is the most important one and the other components are discussed in other area. In our system, we propose the Multi-layer CAI architecture to construct the CAI systems, and use the management of components to distribute the resources over the servers on Internet to achieve the goal of resource sharing.

We present a 3-layer CAI architecture model that expresses different points of view and is fully flexible and component oriented [2,3]. Based on the efficiency of systems, the model is partitioned into 3 layers--resource layer, presentation layer and evaluation layer. It raises the productivity of system development and improvement process, also promotes the individual skills and development of distributed computing environment.

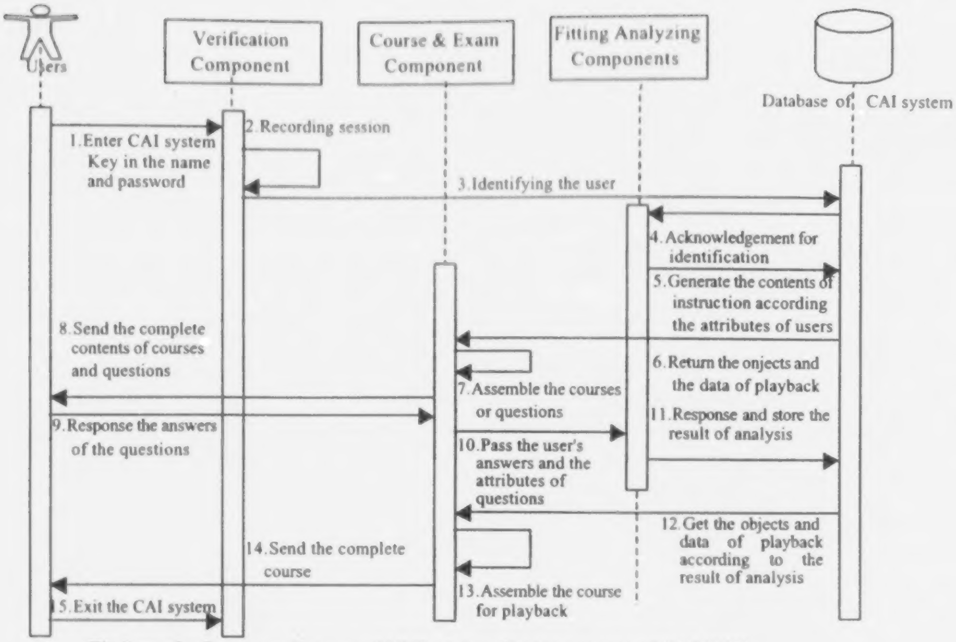

Users
Verification Component
Course & Exam Component
Fitting Analyzing Components
Database of CAI system
1.Enter CAI system Key in the name and password
2.Recording session
3.Identifying the user
4.Acknowledgement for identification
5.Generate the contents of instruction according the attributes of users
6.Return the onjects and the data of playback
7.Assemble the courses or questions
8.Send the complete contents of courses and questions
9.Response the answers of the questions
10.Pass the user's answers and the attributes of questions
11.Response and store the result of analysis
12.Get the objects and data of playback according to the result of analysis
13.Assemble the course for playback
14.Send the complete course
15.Exit the CAI system

Fig.3 use the Sequence diagram of UML to describe the process of CAI [8][9]

3.3 Relationship between Three-layer CAI architecture and hypermedia

From the Table 1 and the frame and resource objects defined in our multimedia presentation system, we can analyze that to what layer the settings of various objects belong listed in Table 2[2][4]. In the components of scenario, we define the miFi that describes which frame should be displayed after the message is triggered, i.e. we can use the expressions to define the schedule of playback of the frames about designing exams and teaching. The components of plot or story just describe the flow of teaching courses defined by users.

From Table 3, we can design and implement the system on Internet more easily to let teachers or other education experts design their teaching materials or questions of exams conveniently and systematically.

4 Conclusion

Different researchers can benefit from this architecture by studying their own knowledge domain independently. Shortening the time spent on completely developing the whole system is to promote the successful rate of resolving the kernel problems. Researchers can' t benefit from studying their own domain only; it's necessary for them to know our open architecture that can easily expand one system into various domains.

Users can acquire an easy-used and reusable system from defining components of multimedia and instructive units of CAI. Our architecture lets teachers have the suitable flexibility and lets various experts and scholars participate in the installation of CAI system. The educational authorities can take our architecture as a referenced architecture for developing the multimedia education. Our system is shown in Fig. 4. The prototype of our system has been completely implemented and published in some various conferences or journals. [1] [5] [9] [10]

Table 1. Three-layer CAI architecture [2][4]

Layer	Researcher	Participants
Resource	Researcher of Interface	Designer of animation , graphic, sound
Presentation	Researcher of learning theory	Teacher, Instructor
Evaluation	Researcher of evaluation	Manager, researcher of educational policy

Table 2. Explanation of part of components [2][4]

First layer (Evaluation layer)	
Components of fitting analysis	This component is made according to some theorem. After analyzing the data acquired from the process that the students take exams and learn, there are some various frames generated.
Components of evaluation and analysis	This component is made according to learning evaluation and learning retrieval of theorist or researchers.
Second layer (Presentation layer)	
Components of scenario	This component is made according to the researchers of learning theory or teaching materials.
Components of structure	This component is made according to learning environment.
Third layer (Resource layer)	
Components of exam	This part must include the parameter or properties which is used broadly
Components of background	Background is concerned to the interest and attention of learner.
Components of referents	To help users of different levels from different method and presentation
Components of multimedia	The components make the CAI lively which may be somebody of cartoon

Table 3. Explanations of part of components

Explanation of part of components		Set the values of necessary item needed
First layer (Evaluation layer)	Components of fitting analysis	
	Components of evaluation and analysis	• F_i.O • F_i.Layout and F_i.UM= exam • F.Presentation and F_i.UM= exam
Second layer (Presentation layer)	Components of scenario	• S
	Components of structure	• F_i.Layout and F_i.UM=learn • F.Presentation and F_i.UM=learn
Third layer (Resource layer)	Components of exam	• O_i.UM = exam
	Components of background	• O_i.UM =Background.
	Components of referents	• O_i.UM = Referents

References

[1] Bal Wang , Ching-Fan Chen, Min-Huei Lin. " A Multimedia Presentation System on Web -- Dynamic Homepage Approach"., The Institute of Electronics, Information and Communication Engineers(IEICE) Transactions on Information and System, Vol.E82-D,NO.4,April 1999. pp. 729~736

[2] Min-Huei Lin, Ching-Fan Chen, Ian-Ru Li, " Building Component Oriented Multi-tier CAI Architecture in the Internet ", ICCE' 98 (The Sixth International Conference on Computers in Education) , Beijing,china,1998,pp.180~pp.186.

[3] Bal Wang , Ching-Fan Chen, Min-Huei Lin , Wei-Yu Li , Ya-Ting Chuang. "A Three-Tier Multimedia Presentation System in Internet", ISMIP 98(International Symposium on Multimedia Information Processing),Taiwan, pp. 162-166.

[4] Bal Wang , Ching-Fan Chen, Min-Huei Lin , Ian-Ru Li. "Building multimedia component oriented Multi-tier CAI architecture in the Internet".ISMIP 98(International Symposium on Multimedia Information Processing),Taiwan, pp. 158-161.

[5] Min-Huei Lin, Bal Wang, Ching-Fan Chen, Chien-Ming Pen , Zheng-Kui Huang , Ian-Ru Li. "An Authoring Tool for Teaching Materials on Web- Dynamic Homepage Approach using SMIL". ICCE' 99(International Conference on Computers in Education).

BEST COPY AVAILABLE

[6] Min-Huei Lin, Chien-Jen Huang, Shang-Zong You, Bal Wang, Ching-Fan Chen ."Using Adjustable Multimedia Honepage Design–A Scheme to Promote the Effect of Web CAI" . ICCE' 99(International Conference on Computers in Education).

[7] Chung, C.-M.; Shih, T.K.; Kuo, C.-H.; Lin, W.-C. . "Supporting the reuse of multimedia presentations" , Information Sciences, Vol: 101, pp. 131-151.

[8] Min-Huei Lin, Zheng-Kui Huang, Bal Wang, , Ching-Fan Chen• "Using Object Oriented Method in Network CAI Model" • The 8th International Conference on Computer-Assisted Instruction.

[9] Min-Huei Lin, Zheng-Kui Huang, Bal Wang , Ching-Fan Chen.• "Management of the Object–Oriented Resources Application System – an Instance of Distance Education" • Distance Education. Vol. 12 , June 1999. pp.53-61.

[10] Bal Wang, , Ching-Fan Chen , Min-Huei Lin , Chien-Ming Pan , Jen-Shan Huang , Hsing-Ten Chein• "An Authoring Tool on Web" • •The 8th International Conference on Computer-Assisted Instruction.

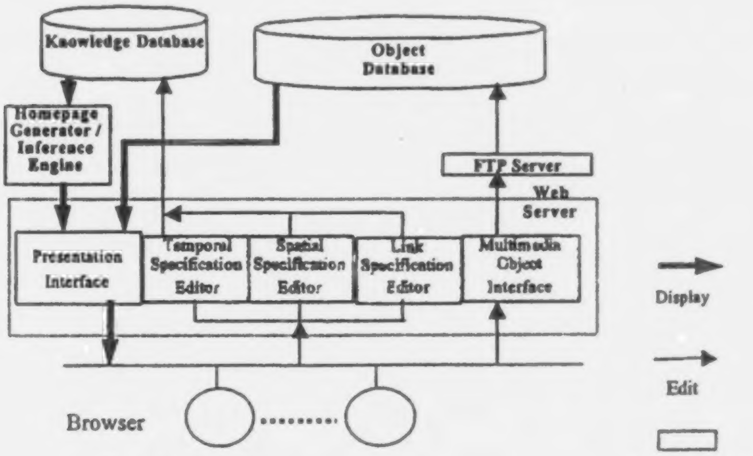

Knowledge Database
Object Database
Homepage Generator / Inference Engine
FTP Server
Web Server
Presentation Interface
Temporal Specification Editor
Spatial Specification Editor
Link Specification Editor
Multimedia Object Interface
Browser
Display
Edit
User Interface

Fig.4. System architecture

CAI System Generator on Web -- using Automatic Trace Recording

Min-Huei Lin, Ching-Fan Chen, Si-Hao Hu, Ming-Hong Taai
and Yung-Hsiang Chiu
Department of Information Management
Aletheia University
Tamsui, Taiwan
R.O.C
Email: lmh@email.au.edu.tw, cfchen@email.au.edu.tw

By the prosperity of computer media, many companies treat electric media as their developmental base and use these electric media in more effective way. It goes without saying that the domain of teaching has developed on the Internet and many CAI systems have been already used in the teaching. The goal of our research is to create CAI systems by automatically recording the trace of editing. So in the thesis, we define the actions of users through image, audio, schedule, point and the module of event, and present the generated CAI systems dynamically on web.

Keywords: **CAI System Generator, Multimedia, Web**

1 The goal

Currently, many teachers and students use CAI systems as their teaching tools, and most teaching materials are designed by both teachers and system engineers. But teachers are generally in the passive position, and if they want to make teaching materials according to their own ideals, they have to learn how to use HTML to design homepages. Usually, students may not understand the meanings of teaching materials very well through the static homepages written in HTML. So we propose and implement an auto-recorded multimedia presentation system to let authors construct dynamic homepages of CAI systems directly through browser on web from automatically recording the trace of their editing.

2 Structure of system

We show the structure of our system in Fig. 1. In the auto-recorded system, we can catch the screen of process of users' operations, or insert sound or image information to the process. Then, these multimedia resources and related information are stored in Information Database and Media Database. The information of presentation schedule is recorded in information database. In the media database, contents of multimedia objects are recorded. In Fig.2, we can see the interactions among Image, Sound, Timer, pointer and Event. Image Module is to make necessary pick-ups for required images, decide what images are picked up in the Event Module Database and store their transition and filename in the forms. Sound Module is used to record sound, thereafter the sounds can be played at proper time by temporal scheduling. Pointer Module is to record the location of mouse pointer. When the transition has something wrong, we can make an adjustment in the coordination. In Timer Module, the time sequences are recorded in the form of Timer Pointer. The schedule designed through directly recording or specified by users is stored in the event database, and the generated multimedia objects will be presented according to the schedule built on the Timer Pointer. Event Module will react to all the other modules. It can decide what modules are going to work, and react to them. When users need to present teaching materials, the Java & HTML Generator will generate and send java code and HTML code to users' browser, then users can see the dynamic homepages. In Fig. 3, we can see a dynamically presented Web CAI system that is produced by recording and modified through the authors' edition and arrangement.

BEST COPY AVAILABLE

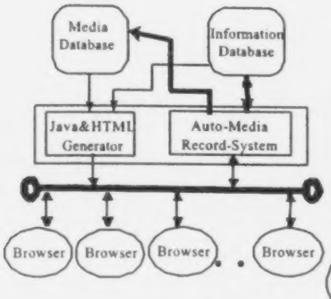

Media Database
Information Database
Java&HTML Generator
Auto-Media Record-System
Browser
Browser
Browser
Browser

Fig.1 Structure of System

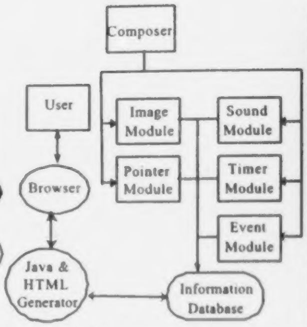

Composer
User
Image Module
Sound Module
Pointer Module
Timer Module
Event Module
Browser
Java & HTML Generator
Information Database

Fig.2 Modules of System

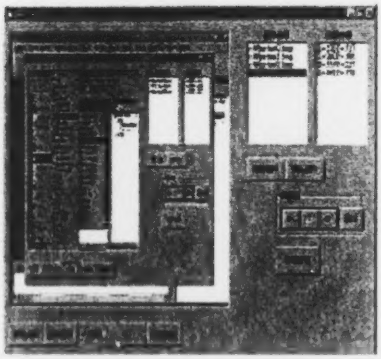


Fig.3 The interface of Auto-Recorded System.

3 Conclusion

We still continuously work on the pack technique of the multimedia file because the transmittance of image and audio are limited by the bandwidth of the Internet. However, teaching through Internet is an inevitable trend in the future, so how to make the best efforts between editing the teaching materials and let the learners learn as efficiently as possible are our goals.

References

[1] Min-Huei Lin, Bal Wang, Ching-Fan Chen, Chien-Ming Pen , Zheng-Kui Huang , Ian-Ru Li. "An Authoring Tool for Teaching Materials on Web- Dynamic Homepage Approach using SMIL". ICCE' 99.

[2] Min-Huei Lin, Chien-Jen Huang, Shang-Zong You, Bal Wang, Ching-Fan Chen ."Using Adjustable Multimedia Homepage Design–A Scheme to Promote the Effect of Web CAI" . ICCE' 99.

[3] James F. Allen "Maintaining Knowledge about Temporal Intervals", Communications of the ACM, Vol. 26, No. 11, 1983.

[4] http://windowsmedia.com

CoCoAJ: Supporting Online Correction of Hypermedia Documents for CALL

Hiroaki Ogata, Yoshiaki Hada, Cheng Feng and Yoneo Yano
Dept. of Information Science and Intelligent Systems,
Faculty of Engineering, Tokushima University
2-1, Minamijosanjima, Tokushima 770-8506, Japan
Tel: +81-886-56-7498 Fax: +81-886-23-2761
E-mail: {ogata, hada, feng, yano}@is.tokushima-u.ac.jp
URL: http://www-yano.is.tokushima-u.ac.jp/ogata/

This paper describes an online markup-based composition learning environment system called **CoCoAJ** (Communicative Collection Assisting System for Java). It allows students and teachers to exchange marked-up documents via Internet, and its environment is very similar to a real world one in which people use pen and paper. In order to record and exchange corrected compositions with marks and comments, this paper proposes XCCML (eXtensible Communicative Correction Mark-up Language), that is based on XML (eXtensible Mark-up Language). XCCML facilitates teachers to analyze and reuse the marked-up documents for the instruction.

Keywords: **Computer assisted language learning, Collaborative writing, XML, Online document correction, Hypermedia.**

1 Introduction

Recently, teacher-centered instructional approaches in traditional writing classrooms are replaced with more active and learner-centered learning approaches with collaborative writing tools[2]. These tools can (1) change the way students and teachers interact; (2) enhance collaborative learning opportunities; (3) facilitate class discussion; and (4) move writing from solitary to more active and social learning. Writing compositions includes various sub-processes such as planning, transcribing, and revising, which do not need to occur in any fixed order [19]. In particular, the review process assisted with computer-based writing tools, has recently received much interest (see as examples [4, 9]).

Many researchers developed online markup systems employing some markup models. However, it is very difficult to analyze and reuse the marked documents that are collected through the writing classroom because the documents do not have a common structure. Therefore, it is necessary to define the generalized format for encoding and exchanging the marked-up documents in order that online markup systems are used easily and widely.

CoCoA (Communicative Correction Assistant system) has been developed for supporting foreigners and teachers to exchange marked-up documents by e-mail [14]. Its environment is very similar to a real one in which people use paper and pen. CoCoA allows teachers not only to correct the compositions sent from foreigners by E-mail, but also foreigners to see where and why the teacher had corrected them. CoCoA improves the opportunities that foreigners have for writing Japanese compositions and for receiving instructions from teachers. **CCML** (Communicative Correction Mark-up Language) [15] has also been proposed for the representation of marked-up documents, which is based on **SGML** (Standard Generalized Mark-up Language) [8]. With CCML, teachers and students can exchange marked-up documents via e-mail [16, 17]. In the experimental use of CoCoA, most of users commented that CoCoA was easy for them to understand the mistakes in documents because of the use of marks, and that the optional view of the original, marked or revised text was very useful. However, CoCoA cannot show users a hypermedia document including figures, tables, movies and links because it deals with only text.

This paper tackles how to correct hypermedia documents by the extension of CoCoA. This paper proposes **CoCoAJ** (CoCoA for Java) to do so. Also this paper describes **XCCML** (eXtensible CCML) for correcting

BEST COPY AVAILABLE

hypermedia documents, that are based on XML (eXtensible Markup Language). XCCML is combined CCML with HTML (Hyper Text Markup Language) that can represent hypermedia documents including pictures, movies, audios and so on.

We have been investigating technological support for Japanese language learning among overseas students. For example, CAI systems called Kanji Laboratory [7], JUGAME [23], GRACILE[23] and JULLIET[1] were developed to support Japanese language learning. However, an on-line mark-up supporting system for Japanese language learning has not yet been proposed. Usually, in a Japanese writing classroom, teachers have to individually review learners' documents using pen and paper[18]. It takes a lot of time for teachers to do this. Therefore, we have implemented CoCoA for writing Japanese composition.

2 Online markup models

There are some editing systems that support teachers to review and correct the students' drafts with online mark-up. Farkas & Poltrock [5] classified the mark-up models as followings:

(1) *Silent editing model*: This is the simplest model and it requires no special techniques. However, it is very difficult for the author to check the editor's work. This model is destructive because the editor cannot readily recover the original words once he/she has changed it.

(2) *Comment model*: This model employs pop-up notes, temporary footnotes, hidden text, and special symbols placed within the text. This model can work for special groups and ad-hoc situations. A system called XyWrite[10] was proposed with this model.

(3) *Edit trace model*: In this model, the editor works in the manner of an author, deleting, adding, and moving text as usual. The computer can compare the editor's new version with the original text, and allows the author to view the draft that contains the editor's changes. This model is apt to encourage heavier editing and less regard for the author's original text. Microsoft Word accepts this model.

(4) *Traditional mark-up model*: This adapts the traditional paper mark-up model to the computer screen. The symbols are both familiar and intuitive for editors and authors; for example, deletion, insertion, and move. For instance, Red Pencil allows the editor to apply a complete set of traditional editing symbols directly to a document. The editor uses "digital ink" to mark a traditional editing symbol along with the words. Moreover, MATE[6] allows the editors to use both digital ink and voice command toward pen and voice computing. In this model, authors and editors can interpret the editor's markings much more readily than in the edit trace model.

There are many systems that employ traditional mark-up which allows multiple users to mark-up an electronic document as if they were marking up a printed copy of the document. However, such systems do not globally come into practical and wide use in composition writing classes because of their special format. Moreover, it is very difficult to analyze and reuse the marked documents because the marked documents are unstructured. Therefore, the system should provide a generalized and structured format for encoding and interchanging marked-up documents via the Internet.

3 XCCML

Based on the experimental results, we propose XCCML for exchanging marked-up documents. XCCML is an application of XML, and it supplies a formal notation for the definition of generalized mark-up languages. XML is a device- and system-independent method of representing texts in electronic form. That is to say, XML is a set of mark-up conventions used together for encoding texts. A mark-up language must specify what mark-up is allowed, what mark-up is required, how mark-up is to be distinguished from text and what the mark-up means.

3.1 Features of XCCML

The main characteristics of XCCML are:

(1) Based on the experiment, XCCML presents six marks and annotation XCCML tags.

(2) The marks have three degrees of importance levels against respective corrections.

(3) The original text is generated through removing all the XCCML tags.

(4) The revised text is derived from the XCCML document.

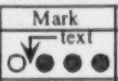
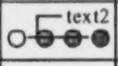
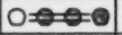
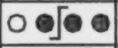
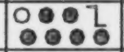
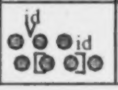
BEST COPY AVAILABLE

(5) Because XCCML documents are text-formatted, it is easy to send them by e-mail.
(6) CCML documents easily make up full-text databases.
Needless to say, XCCML inherits its features from XML.

3.2 XCCML structure

As shown in table 1, XCCML documents consist of three parts: header, body and close. "Header" represents additional information about the document. For instance, "next" tag denotes the next version of the document. The marks for review are shown in the "Body" as XCCML tags. "Close" shows the editor's comments. In one sentence, "insert," "replace" and "delete" marks were used, while "join," "separate" and "move" marks were used over two sentences. The part between the start tag and the end tag denotes the learner's mistakes. The "string" attribute represents the revised part of the document.

Table 1: Marks and XCCML tags.

Correction	Mark	Tag
1. Insert		<Insert string="text"/>
2. Replace		<Replace string="text2"> text </Replace>
3. Delete		<Delete> text </Delete>
4. Separate		<Separate/>
5. Join		<Join/>
6. Move		<Movefrom refid="id"/> <Moveto id="id"> text </Moveto>

(1) Root tags

Tag name	Explanation	Attribute	Attribute's contents	End tag
XCCML	Beginning of XCCML tag	Version	Version number	</XCCML>
Head	Header information	None		</Head>
Body	Corrected document	None		</Body>
Close	Overall comments	None		</Close>

(2) Tags in header section

Tag name	Explanation	Attribute	Attribute's contents	End tag
Title	Title of the document	String	Title name	None
Editor	People who corrected the document	Name	Name of the editor	None
		Email	Email address	
Author	People who write the original document	Name	Name of the author	None
		Email	Email address	

(3) Tags in body section

Tag name	Explanation	Attribute	Attribute's contents	End tag
Insert	Insert words	String	Inserted words	</Insert>
		Level	Level of importance	
		Comment	Comment for the correction	
Replace	Change words	String	Corrected words	</Replace>
		Level	Level of importance	
		Comment	Comment for the correction	
Delete	Delete words	Level	Level of importance	None
		Comment	Comment for the correction	
Separate	Separate a paragraph	Level	Level of importance	None

CoCoAJ: Supporting Online Correction of Hypermedia Documents for CALL

Hiroaki Ogata, Yoshiaki Hada, Cheng Feng and Yoneo Yano
Dept. of Information Science and Intelligent Systems,
Faculty of Engineering, Tokushima University
2-1, Minamijosanjima, Tokushima 770-8506, Japan
Tel: +81-886-56-7498 Fax: +81-886-23-2761
E-mail: {ogata, hada, feng, yano}@is.tokushima-u.ac.jp
URL: http://www-yano.is.tokushima-u.ac.jp/ogata/

This paper describes an online markup-based composition learning environment system called **CoCoAJ** (Communicative Collection Assisting System for Java). It allows students and teachers to exchange marked-up documents via Internet, and its environment is very similar to a real world one in which people use pen and paper. In order to record and exchange corrected compositions with marks and comments, this paper proposes XCCML (eXtensible Communicative Correction Mark-up Language), that is based on XML (eXtensible Mark-up Language). XCCML facilitates teachers to analyze and reuse the marked-up documents for the instruction.

Keywords: **Computer assisted language learning, Collaborative writing, XML, Online document correction, Hypermedia.**

1 Introduction

Recently, teacher-centered instructional approaches in traditional writing classrooms are replaced with more active and learner-centered learning approaches with collaborative writing tools[2]. These tools can (1) change the way students and teachers interact; (2) enhance collaborative learning opportunities; (3) facilitate class discussion; and (4) move writing from solitary to more active and social learning. Writing compositions includes various sub-processes such as planning, transcribing, and revising, which do not need to occur in any fixed order [19]. In particular, the review process assisted with computer-based writing tools, has recently received much interest (see as examples [4, 9]).

Many researchers developed online markup systems employing some markup models. However, it is very difficult to analyze and reuse the marked documents that are collected through the writing classroom because the documents do not have a common structure. Therefore, it is necessary to define the generalized format for encoding and exchanging the marked-up documents in order that online markup systems are used easily and widely.

CoCoA (Communicative Correction Assistant system) has been developed for supporting foreigners and teachers to exchange marked-up documents by e-mail [14]. Its environment is very similar to a real one in which people use paper and pen. CoCoA allows teachers not only to correct the compositions sent from foreigners by E-mail, but also foreigners to see where and why the teacher had corrected them. CoCoA improves the opportunities that foreigners have for writing Japanese compositions and for receiving instructions from teachers. **CCML** (Communicative Correction Mark-up Language) [15] has also been proposed for the representation of marked-up documents, which is based on **SGML** (Standard Generalized Mark-up Language) [8]. With CCML, teachers and students can exchange marked-up documents via e-mail [16, 17]. In the experimental use of CoCoA, most of users commented that CoCoA was easy for them to understand the mistakes in documents because of the use of marks, and that the optional view of the original, marked or revised text was very useful. However, CoCoA cannot show users a hypermedia document including figures, tables, movies and links because it deals with only text.

This paper tackles how to correct hypermedia documents by the extension of CoCoA. This paper proposes **CoCoAJ** (CoCoA for Java) to do so. Also this paper describes **XCCML** (eXtensible CCML) for correcting hypermedia documents, that are based on XML (eXtensible Markup Language). XCCML is combined CCML with HTML (Hyper Text Markup Language) that can represent hypermedia documents including pictures, movies, audios and so on.

We have been investigating technological support for Japanese language learning among overseas students. For example, CAI systems called Kanji Laboratory [7], JUGAME [23], GRACILE[23] and JULLIET[1] were developed to support Japanese language learning. However, an on-line mark-up supporting system for Japanese language learning has not yet been proposed. Usually, in a Japanese writing classroom, teachers have to individually review learners' documents using pen and paper[18]. It takes a lot of time for teachers to do this. Therefore, we have implemented CoCoA for writing Japanese composition.

2 Online markup models

There are some editing systems that support teachers to review and correct the students' drafts with online mark-up. Farkas & Poltrock [5] classified the mark-up models as followings:

(1) *Silent editing model*: This is the simplest model and it requires no special techniques. However, it is very difficult for the author to check the editor's work. This model is destructive because the editor cannot readily recover the original words once he/she has changed it.

(2) *Comment model*: This model employs pop-up notes, temporary footnotes, hidden text, and special symbols placed within the text. This model can work for special groups and ad-hoc situations. A system called XyWrite[10] was proposed with this model.

(3) *Edit trace model*: In this model, the editor works in the manner of an author, deleting, adding, and moving text as usual. The computer can compare the editor's new version with the original text, and allows the author to view the draft that contains the editor's changes. This model is apt to encourage heavier editing and less regard for the author's original text. Microsoft Word accepts this model.

(4) *Traditional mark-up model*: This adapts the traditional paper mark-up model to the computer screen. The symbols are both familiar and intuitive for editors and authors; for example, deletion, insertion, and move. For instance, Red Pencil allows the editor to apply a complete set of traditional editing symbols directly to a document. The editor uses "digital ink" to mark a traditional editing symbol along with the words. Moreover, MATE[6] allows the editors to use both digital ink and voice command toward pen and voice computing. In this model, authors and editors can interpret the editor's markings much more readily than in the edit trace model.

There are many systems that employ traditional mark-up which allows multiple users to mark-up an electronic document as if they were marking up a printed copy of the document. However, such systems do not globally come into practical and wide use in composition writing classes because of their special format. Moreover, it is very difficult to analyze and reuse the marked documents because the marked documents are unstructured. Therefore, the system should provide a generalized and structured format for encoding and interchanging marked-up documents via the Internet.

3 XCCML

Based on the experimental results, we propose XCCML for exchanging marked-up documents. XCCML is an application of XML, and it supplies a formal notation for the definition of generalized mark-up languages. XML is a device- and system-independent method of representing texts in electronic form. That is to say, XML is a set of mark-up conventions used together for encoding texts. A mark-up language must specify what mark-up is allowed, what mark-up is required, how mark-up is to be distinguished from text and what the mark-up means.

3.1 Features of XCCML

The main characteristics of XCCML are:

(1) Based on the experiment, XCCML presents six marks and annotation XCCML tags.

(2) The marks have three degrees of importance levels against respective corrections.

(3) The original text is generated through removing all the XCCML tags.

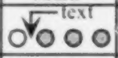
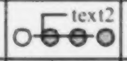
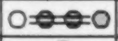
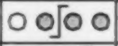
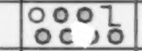
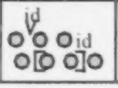
(4) The revised text is derived from the XCCML document.

(5) Because XCCML documents are text-formatted, it is easy to send them by e-mail.
(6) CCML documents easily make up full-text databases.
Needless to say, XCCML inherits its features from XML.

3.2 XCCML structure

As shown in table 1, XCCML documents consist of three parts: header, body and close. "Header" represents additional information about the document. For instance, "next" tag denotes the next version of the document. The marks for review are shown in the "Body" as XCCML tags. "Close" shows the editor's comments. In one sentence, "insert," "replace" and "delete" marks were used, while "join," "separate" and "move" marks were used over two sentences. The part between the start tag and the end tag denotes the learner's mistakes. The "string" attribute represents the revised part of the document.

Table 1: Marks and XCCML tags.

Correction	Mark	Tag
1. Insert		<Insert string="text"/>
2. Replace		<Replace string="text2"> text </Replace>
3. Delete		<Delete> text </Delete>
4. Separate		<Separate/>
5. Join		<Join/>
6. Move		<Movefrom refid="id"/> <Moveto id="id"> text </Moveto>

(1) Root tags

Tag name	Explanation	Attribute	Attribute's contents	End tag
XCCML	Beginning of XCCML tag	Version	Version number	</XCCML>
Head	Header information	None		</Head>
Body	Corrected document	None		</Body>
Close	Overall comments	None		</Close>

(2) Tags in header section

Tag name	Explanation	Attribute	Attribute's contents	End tag
Title	Title of the document	String	Title name	None
Editor	People who corrected the document	Name	Name of the editor	None
		Email	Email address	
Author	People who write the original document	Name	Name of the author	None
		Email	Email address	

(3) Tags in body section

Tag name	Explanation	Attribute	Attribute's contents	End tag
Insert	Insert words	String	Inserted words	</Insert>
		Level	Level of importance	
		Comment	Comment for the correction	
Replace	Change words	String	Corrected words	</Replace>
		Level	Level of importance	
		Comment	Comment for the correction	
Delete	Delete words	Level	Level of importance	None
		Comment	Comment for the correction	
Separate	Separate a paragraph	Level	Level of importance	None

		Comment	Comment for the correction	
Join	Join paragraphs	Level	Level of importance	None
		Comment	Comment for the correction	
Moveto	Moved position from the original spot	Fromid	ID	None
		Comment	Comment for the correction	
Movefrom	Original position of the words	Toid	ID	</Movefrom>
		Level	Level of importance	
Annotate	Comment for the correction	String	Annotation contents	None
		Level	Level of importance	

(4) Tags in close section

Tag name	Explanation	Attribute	Attribute's contents	End tag
Comment	Overall comment	None	None	Necessary

3.3 Level of marks

We found that the marks do not have the same level of importance. We identify corrections on the following levels:

(1) Weak correction: The learner does not need to revise the document.
(2) Normal correction: The learner should correct the document.
(3) Strong correction: The learner must correct the document.

The strong corrections denote the important part of marks to be revised in the document. Using the importance level that the teacher had entered, the system provides the learner with the marks he/she wants to see. Therefore, the learner can avoid information overload from the reviewed documents. Every tag in table 1 has an attribute "level" that a teacher gives a number from one to three. Its default is two as normal correction.

3.4
Level of annotations

It is very important for a teacher to annotate the marked text for instruction in composition. For example, PREP Editor [12] is a word processor that allows writers and reviewers to create electronic margins, or columns, in which they can write and communicate through their annotations. We identify the following different kinds of annotations:

(1) Explanation: This is used for explaining the reason of a correction.
(2) Question: This is used for asking the learner a question; e.g., what do you want to write?
(3) Comment: This shows the educational view of the teacher with respect to the document.

4 CoCoAJ

CoCoAJ consists of CoCoAJ-Editor and CoCoAJ-Viewer. A teacher corrects the document of overseas students with CoCoAJ-Editor and the system saves the marked document in XCCML format. CoCoAJ-Viewer interprets the XCCML document, and shows the learner the marked document.

4.1 Learning processes using CoCoAJ

By using CoCoAJ, a learner receives instruction about a Japanese composition from a teacher with the following processes:

(1) The learner writes an original text with his/her familiar editor.
(2) The learner sends the document to his/her teacher with his/her own e-mail tool.
(3) CoCoAJ-Editor makes the document double-spaced. The teacher corrects the document with online marks and annotations. Then, the system allows the teacher to set the importance level to the marks in the document.
(4) After CoCoAJ-Editor saves the marked text as a XCCML document, the teacher sends it to the learner by e-mail.

(5) CoCoAJ-Viewer provides the learner with the marked text after interpreting the XCCML document. Then, the system allows the learner to select the importance level to see the important part of the marked text.
(6) CoCoAJ-Viewer automatically generates both the original text and the revised one from the XCCML document. After editing the revised text, the learner can send it again to the teacher and continue refining the text.
(7) CoCoAJ maintains the version of the document, if the learner wants to revise the same document.

4.2 System configuration

Figure 1 depicts the learning environment of CoCoAJ.

(1) XCCML parser: This module analyzes XCCML documents using the XCCML parser after reading them through the file management module. Then, it provides the results of correction according to the level of importance of marks.
(2) Correction module: This module inserts XCCML tags into the learner's document, according to the revision of the teacher. After saving the marked text, the teacher sends it by e-mail to the learner.
(3) Original text display module: This module generates the original text from the XCCML document by removing all the XCCML tags.
(4) Revised text display module: This module generates the revised text by applying XCCML tags.
(5) File management module: This module manages the versions of the documents. When the learner sends the teacher the revised document, the system creates a new XCCML document, inserts the "next" tag into the old XCCML document, and also enters the "previous" tag into the new XCCML document.

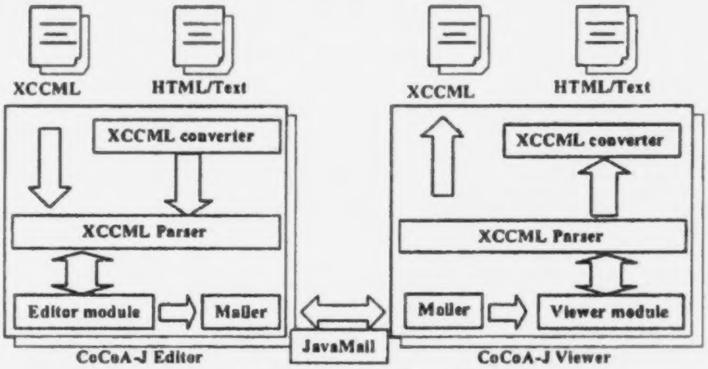


Figure 1: System configuration of CoCoAJ.

4.3 User interface

Figure 2 shows the screen snapshot of CoCoAJ-Editor. First, the learner writes a Japanese composition with a word processor and saves the document as HTML format. After that, the learner sends the document to the teacher by e-mail. By selecting a mark from the mark palette shown in the upper window, the teacher can revise the document. Moreover, the teacher can annotate the document using the annotation palette, and he/she can classify the marks according to the level of importance. The user can see the correcting document at the left side in the window and "*" means the user inserted the comment. The user can see the comments for the correction at the right side in the window. In this figure, the teacher substitutes "allow" with "allows" and gives a comment "*2". Also the teacher can see the original document and revised one by selecting window tag. After saving the marked document as a XCCML (see appendix A), the teacher can send it to the student by e-mail. Using CoCoAJ-Viewer, the learner obtains the same marked text that the teacher revised. By selecting the level of importance, CoCoAJ-Viewer provides only the marks over the level. The learner can reply to the teacher's comments and collaboratively write a composition with the teacher.

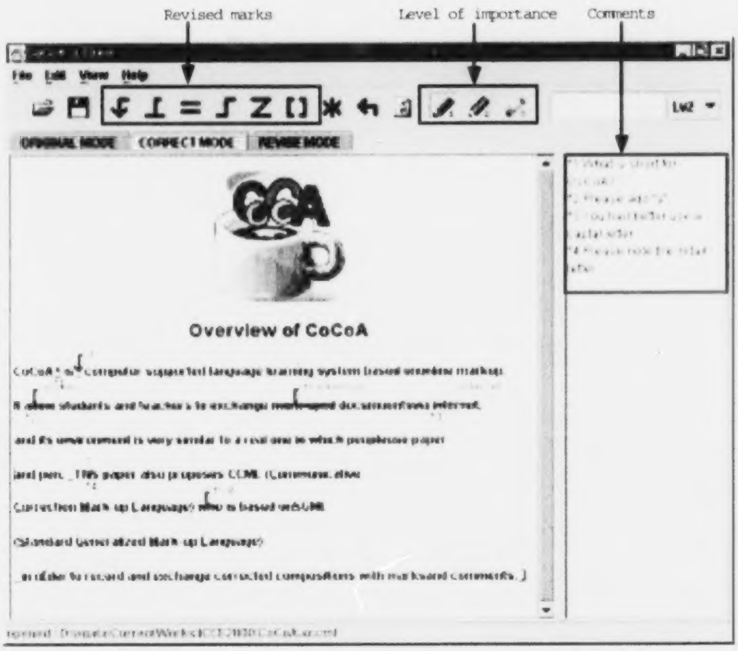


Figure 2: Screen snapshots of CoCoAJ-Editor.

5 Conclusions

This paper proposed a computer mediated language-learning system called CoCoAJ and XCCML for exchanging electronic marked-up documents. Now we are trying to propose XCCML to W3C (World Wide Web Consortium), and to show an XCCML document into Web browsers. After that, CoCoAJ will be able to be used for learning any language in an open-ended writing classroom. In our future research, we will investigate how to classify students' writing errors in their drafts, and how to assist a review process with AI technologies.

Acknowledgment

This research was supported in part by the Grant-in-Aid for Scientific Research (B) (2) No.09558017, No.09480036, and No.09230214 from the Ministry of Education, Science, Sports and Culture in Japan.

References

[1] Ayala, G. & Yano, Y. (1995). "GRACILE: A framework for collaborative intelligent learning environments", *Journal of the Japanese Society of Artificial Intelligence*, vol.10, no. 6, pp.156-170.

[2] Bonk, J. C. & King, S. K. (1995). "Computer conferencing and collaborative writing tools: Starting a dialogue about student dialogue," 1995 Proc. of Computer Supported Collaborative Learning.

[3] Brock, N. M. (1995). "Computerised text analysis: Roots and research," *Computer Assisted Language Learning*, vol.8, no.2-3, pp.227-258.

[4] Coniam, D. (1997) "A computerised English language proofing cloze program," *Computer Assisted Language Learning*, vol. 10, no. 1, pp.83-97.

[5] Farkas K. D., & Poltrock, E. S. (1995). "Online editing mark-up models, and the workspace lives of editors and writers," *IEEE Trans. in Professional Communication*, 38(2), pp.110-117.

[6] Hardock, G., Kurtecbach, G., & Buxton, W. (1993). "A marking based interface for collaborative writing," *Proc. of the ACM Symposium on User Interface Software Techonogy*, pp.259-266.

[7] Hayashi, T., & Yano, Y. (1994). "Kanji Laboratory: An environmental ICAI system for Kanji learning," *IEICE Trans. on Information and Systems*, E77-D(1), pp.80-88.

[8] Herwijnen, E. (1990). *Practical SGML*, Kluwer Academic Publishers.

[9] Kehagia, O. & Cox, M. (1997). "Revision changes when using word processors in an English as foreign languages context," *Computer Assisted Language Learning*, vol. 10, no. 3, pp.239-253.

[10] Kincade, D. & Oppenheim, L (1994) "Marking it up as we go along: Into editorial production's electronic future," *Journal of Scholarly Publishing*, Vol. 25, No. 3, pp.233-242.

[11] Matsumoto, Y, Kitauchi, A., Yamashita, T., Hirano, Y., Imaichi, O. & Imamura, T. (1997) "Japanese Morphological Analysis System ChaSen Manual," Nara Institute of Science and Technology Technical Report NAIST-IS-TR 97007. (in Japanese)

[12] Neuwirth, M. C., & Wojahn, G. P. (1996). "Learning to write: Computer support for a cooperative process," *CSCL: Theory and practice of an emerging paradigm*, Koshmann, T. (Ed.), Lawrence Erlbaum Associates, Publishers, pp.147-170.

[13] Ochi, Y., Yano, Y., & Hayashi, T. (1996). "JULLIET: Interactive learning environment for Kanji compounds learning," *Proc. of Educational Multimedia and Hypermedia 1996*, pp.539-544.

[14] Ogata, H. and Yano, Y. (1997). CoCoA: Communicative Correction Assisting System for Composition Studies, *Proc. of ICCE 97*, pp.461-468.

[15] Ogata, H., Yano, Y., and Wakita, R. (1998). CCML: Exchanging Marked-up Documents in a Networked Writing Classroom, *International Journal of Computer Assisted Language Learning*, Vol.11, No.2, pp.201-214.

[16] Ogata, H., Feng, C., Hada, Y., and Yano, Y. (1999). Computer Supported Proofreading Exercise in a Networked Writing Classroom, *Proc. of ICCE 99*, Vol.1, pp.414-417.

[17] Ogata, H., Feng, C., Hada, Y. and Yano, Y. (2000). Online Markup Based Language Learning Environment, *International Journal of Computers & Education*, Vol.34, No.1, pp.51-66.

[18] Sato, M., Kano, C., Tanabe, K., & Nishimura, Y. (1994). *Practical Japanese Compositions*, Bonjin-sya. (in Japanese)

[19] Scardamalia, M., & Bereiter, C. (1986). "Research on written composition," M. Wittrock (Ed.), *Handbook of research on teaching* (third ed.), Macmilian, pp.778-801.

[20] Schwind, B. C. (1995) "Error analysis and explanation in knowledge based language tutoring," *Computer Assisted Language Learning*, vol.8, no.4, pp.295-324.

[21] Shimano, H. (1986). *Handbook of proofreading*, Miki Publishers, Japan, in Japanese.

[22] Yano, Y., Miyoshi, K., & Hayashi, T. (1993). "Development of a game style environmental ICAI system for Kanji idiom learning," *Proc. of International Conference on Computers in Education '93*, pp.411-413.

[23] Yano, Y., Ogata, H., Sakakibara, R., & Wakita R. (1997). "CoCoA: Communicative correction assisting system for learning Japanese compositions", *Transactions of Japanese Society for Information and Systems in Education*, vol. 14, no.3, pp.21-28. (in Japanese)

Appendix A: **XCCML document in figure 4.**

```xml
<?xml version="1.0" encoding="Shift_JIS"?>
<!DOCTYPE XCCML SYSTEM "XCCML.dtd">
<XCCML>
<HEAD>
<Title string="Overview of CoCoA"/>
<Editor name="Hiroaki Ogata" email="ogata@is.tokushima-u.ac.jp"/>
<Author name="Yoshiaki Hada" email="hada@is.tokushima-u.ac.jp"/>
</HEAD>
<BODY>
<CENTER><IMG width="128" height="128"
src="image001.gif"/></CENTER><CENTER><H2>Overview of CoCoA</H2></CENTER> <H4>
CoCoA <Annotate level="3" comment="What is short for CoCoA?"/> is<Insert string="a" level="3"/>
computer supported language learning system based ononline markup.<BR/>
 It<Replace string="allows" level="3" comment="Please add &#34;s&#34;."> allow </Replace> students
and teachers to exchange<Replace string="marked-up" level="3"> mark-uped</Replace> document via
<Replace string="Internet" level="1" comment="You had better use a captal letter.">
internet</Replace>,<BR/> and its environment is very similar to a real one in which people use paper<BR/>
and pen. <MoveTo fromid="1" level="2" comment="Please note the initail letter."/>This paper also
proposes CCML (<U>C</U>ommunicative<BR/> <U>C</U>orrection <U>M</U>ark-up
<U>L</U>angua<Replace string="that" level="3">ge)</Replace> who is based on SGML<BR/>
 (<U>S</U>tandard <U>G</U>eneralized <U>M</U>ark-up <U>L</U>anguage)<BR/>
<MoveFrom toid="1" level="2">in order to record and exchange corrected compositions with marks and
comments.</MoveFrom></H4></BODY>
<CLOSE>
<Comment/></CLOSE>
</XCCML>
```

Construct in-service Training Web Site for School Teachers

Jia-Rong Jerome Wen* and Wen-Chin Chen**
**jerome@nknucc.nknu.edu.tw*
***timmy@icemail.nknu.edu.tw*

1 Introduction

Information technology grows rapidly recently. People use Internet to obtain many kinds of information. The Internet has become the most important path in cyber world. In the last five years, using the Internet to carry out distance learning, especially for teachers' in-service training, changes the style of education.

To actualize the policy which was to build an lifelong-learning education environment, Ministry of Education delegated National Kaohsiung Normal University (NKNU) to manage Asynchronous Distance Learning class for high school teachers in Oct, 1999.

2 Construct Asynchronous Distance Learning Web page

Generally speaking, teachers have to control the instructive materials, activities, learning process and evaluation. Every instruction system must include all of the three factors as following:

2.1 Instructive materials and activity designing

If we just put the materials onto web site, they look like electronic books on Internet. It is helpless for students. Therefore, when designing the contents of curriculum, we make it in "practicing" orientation. Activities make teacher and students interact with each other and avoid students to feel humdrum or like reading an electronic book.

2.2 Evaluation

When students finish learning a chapter, we give them an formative evaluation to verify whether students master the thesis or not. If students pass the formative evaluation, they can continue the curriculum. If not, they have to go back and learn it again until they pass the formative evaluation. The system would give some feedback to students, they would know which part of contents they don't understand yet. Then, we always hold an examination when finishing the curriculum, the summative evaluation. (Figure-1)

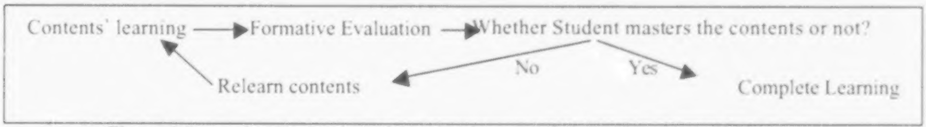


Figure-1 Formative Evaluation of Contents' Learning (Kuo Sheng-Iu,1993,p294)

2.3 Learning process

Instead of quantification of examination, we should care about the reflection from students after instruction and learning. Grades cannot decide students' learning efficiency. During designing the materials, we considered every details of students' learning process. These include

- counts in connection
- counts in joining the forum
- contents what student discuss
- chatting situation between teacher and students

- Homework – students work hard or not
- Chapter evaluation (formative evaluation)
- teachers and students communicate by email

All items can be saved into database so that we can estimate student' s behavior in learning.

3 Concepts on designing curriculum

We design several activities and strategies. The activities will make students concentrate on the contents. we consider about the strategies as following:

3.1 Homework:

We assign homework after students finish learning every chapter. They can evaluate themselves through homework to know how much they learn and review contents again.

3.2 Operative Orientation:

In the homework, they have to work some operation by computer, such as computer game.

3.3 Self-determination:

Self-determination means that students have to study by themselves and plan study schedule by themselves.

3.4 Interactive :

We define interaction into two ways : one direction and two direction (Table-1).

	One direction	Two direction
Asynchronous	Announcement FAQ	On-Line Forum
Synchronous		Chat Room

One direction: teacher to student
Two direction: teacher and student

Table-1 Interactive model

3.5 User Interface• •

What users feel about it is very important in internet environment. Hyperlink always be mazes for a novice in the internet. Trying to solve user interface problem , we use several ways as following :

3.5.1 Frame• cut web pages into several frame to reduce confusing

3.5.2 Tree menu• from the reaction of students , the tree menu is easy to access the pages

3.5.3 Learning Path• guideline for students on the web

References

[1] Fred Jevons."Distance Education and Campus-Based Education : Parity of Esteem", New South Wales, Australia : Peter Smith and Mavis Kelly. (1987)

[2] Kuo Sheng-Iu(1993). "The Evaluation of Psychology and Education", Chin-Hua , 7th Edition., (1993).

[3] Lisa Neal(1998). "Distance Learning". ACM(http://www.acm.org)(accessed on May 15,2000).

[4] Mei-Sheng Chao, "The Impact of a Systems Approach on Interactive Distance Teaching: The Development of the ATM Video Network System at National Chung-cheng University", ETR&D, Vol 47. No 2. pp. 112-117(1998).

[5] Sun Chuen-tzai, Chu Ching-hao, & Shiu Hdi, "The research of constructing hypermedia teaching material", Information and Education• vol. 67• pp 13-25(1998).

[6] Wang Chien-Shin, "Web-based network system for instructive management", Information and Education, Vol. 67• pp 3-12(1998)

[7] Wen Jia-rong, "The research on present Statement and Requirements of junior-high-school Information Education", Information and Education, Vol. 51, pp17-26(1996)

[8] Wen Jia-rong, Wu Ming-lung , "Information Education Theories and Applications in New Age ", Song Kang ,1st Edition.(1999).

[9] Han Shan-ming, "Information Education Fundamental Construction – Speed", Information and Education, Vol. 68, pp14-16 (1998)

From Research to Development: A Content Analysis of Journal Articles

Alex Rath* and Shihkuan Hsu**
** Shih Hsin University, Language Center,1 , Alley 17, Sec. 1, Mucha Rd., Taipei, Taiwan 110*
Tel: +886-2-22368225 x316, E-mail: rattle@cc.shu.edu.tw
*** National Taiwan University, CTE, 1, Sec. 4, Roosevelt Rd., Taipei, Taiwan 106*
Tel: +886-2-23692364 x214, E-mail: skhsu@ccms.ntu.edu.tw

While the goal of educational research is to improve the field of education, most of the material used in the field is not necessarily developed from research. While most of the researchers in the university study the theory and practice, it is the textbook publishers and computer companies that make the tools available for schools and teachers. The disconnected and fragmented process between research, dissemination, development, and production not only wastes the effort and resources put forth in research and practice, but also hampers the development of the field of education. Using the method of content analysis, the authors examine the connectivity of educational research and development reported in a highly regarded professional journal over the past two decades. The results found that over 80% of the research studies presented only parts of a full-fledged R&D process: Theory, research, development, funding, and dissemination. Among them, about 40% of the articles addressed both research and development. It is argued that although the full-fledged R&D is a crucial process to enable the transformation of good research into quality practice, it has not received enough attention in the field of education. Implications for educational technology are particularly discussed.

Key words: **Educational Research, Research and Development, Content Analysis, Instructional Technology, Information Dissemination**

1 Introduction

The separation of research and development has long haunted the field of education. The professional approach to creating new instructional tools in the academic realm starts with educational research. The corporate instructional tool developers reiterate that the process starts with research. Theoretically, the process includes three stages: (1) research and development (also called R&D), (2) product development, and (3) production Dumbleton [1]. In actual practice, very little of the research makes it out of the lab and becomes widely used in practice. At the same time, corporate developers of instructional technology or curriculum material invest little in educational research. To go beyond personal observations and anecdotal evidence, a method is invented to examine the level of full-fledged R&D available in professional journals that are widely circulated in educational circles. The research question was: What amount of full-fledged R&D that includes theory, research, development, funding and dissemination was published in the *American Educational Research Journal*?

2 Method

In this study, the method of content analysis is used to evaluate the level of educational R&D reported in journals. Content analysis, or bibliometric analysis, includes evaluation of macro-trends such as dissemination as well as microanalysis such as the nuance of a single line of text. Ample precedent exists to use content analysis to evaluate the state of educational R&D. Articles in *AERJ* and other journals have been analyzed in order to understand R&D productivity, university productivity, faculty productivity, program quality, methodological quality, journal content, and journal rhetoric. Therefore, analysis of articles in *AERJ* provides a measurable indication of the extent that full-fledged educational R&D is published.

Random selection produced a dataset of *AERJ* articles (N=278). About 40% (n=116) of the papers directly and indirectly related to instruction. Of the articles, 74 indirectly related to instruction and were eliminated. Excluding all articles unless they were directly related to instruction produced a small subset, comprising about 15% (*n*=42) of the sample, which were analyzed in this study. Articles were coded according to five components: Theory, research, development, funding, and dissemination. The range is 1-5, where 1 indicates only one part of the R&D process was included and 5 indicates that all five parts were found in the article. The mean number of R&D components reported in the instruction articles (*n*=42) was 3.36 and the standard deviation was 0.98.

3 Results

Content analysis was used to determine the extent to which theory, research, development, funding, and dissemination appeared in the selected articles. Evaluating the elements separately showed the frequency of R&D components. All 42 articles (100%) presented research. Thirty-one articles (74%) achieved dissemination. Twenty-nine articles (69%) reported funding. Twenty-one articles (50%) discussed theory. Of all the parts of R&D, development was the lowest frequency. Only 18 articles (43%) presented educational development. This was not surprising. Many articles were efforts to see that was happening in classrooms and with individuals, and there were many tests of theory.

The next step in describing the articles on instruction is to determine how many R&D articles combined all five parts. Of the 42 articles on instruction, the complete R&D process was clearly observable in only five (12%) papers. These papers were published in 1964, 1971, 1990, 1996, and 1997, suggesting that the more recent papers have slightly better coverage of the R&D process. Fourteen (33%) of the articles contained four components of R&D. Fifteen (33%) of the articles contained three components. Seven (17%) of the articles contained two components. One article (2%) had just one component.

By combining the two top categories, a new way to interpret the data emerges. The articles in the top two categories, articles with five and four parts of the R&D process, 45% (*n*=19) can be described as more developed studies than the articles with one, two, or three parts. It is interesting to note which component was missing in the articles which reported four parts. The missing component was theory, 42% (*n*=6), development, 42% (*n*=6), funding, 8% (*n*=1), and dissemination, 8% (*n*=1), respectively. There is a pattern emerging: research, funding, and dissemination.

A characteristic paper format emerges which has theory, research, and dissemination. Researchers are testing theories. Combining the three part and four part papers, which includes 69% of the articles (*n*=29), demonstrates the emphasis on educational theory and educational research. The pattern of what is included and what is missing appears when ranked in order of frequency. The four part studies present, in order of frequency, the following components: research, dissemination, funding, theory, and development. The three part studies present, in order of frequency, the following components: research, dissemination, funding, theory, and development. In other words, both three and four parts studies have the same pattern. They are most likely to do research, achieve dissemination, get funding, use theory, and complete educational development. Educational development is last on the list of R&D components presented in nearly seven out of ten *AERJ* articles. When funding and dissemination are dropped, the following pattern emerges: theory and research.

4 Conclusions

The incomplete R&D program is an unfulfilled opportunity. When numerous programs are unfinished, and the results are not fully disseminated, the investment is marginalized. When the knowledge from the incomplete program is not added to the knowledge base, overlap will occur and resources will not be put too good use. Teachers and administrators need choices, and only full-fledged R&D will give them a selection of programs for their schools. Corporate developers, like schools, need educational R&D. With such a limited number of full-fledged R&D articles in *AERJ*, it is possible that education professionals are missing an important opportunity.

References

[1] Dumbleton, J. H., "Management of high-technology research and development", Amsterdam, The Netherlands: Elsevier Science Publishers, 2nd Edition, (1989).

Design and Implementation of a Chinese Web-mail System

Chang-Sheng Chen*, Jian-Liang Liu and Shian-Shyong Tseng****
Department of Computer & Information Science, National Chiao-Tung University
1001, Ta Hsueh Rd., HsinChu, Taiwan 300, R.O.C.
Tel: +886-3-5731721 Ext.56605
Fax: +886-3-5721490
**E-mail: cschen@cc.nctu.edu.tw*
***E-mail: jlliu@cis.nctu.edu.tw, sstseng@cis.nctu.edu.tw*

E-mail is one of the most popular services on Internet. Fast message transportation, good GUI designs and enhanced localization capabilities in end-user environments are the key factors. However, there still exist some addressing problems for many users since it is based primarily on the ASCII character set. For those who do not know English well, ASCII set is hard to memorize and is prone to making errors. As more users joining the Internet, this kind of problems should not be ignored. Especially, these become major problems for students and teachers in primary/secondary schools. Currently, many approaches are proposed to support the Chinese and/or multilingual DNS name resolution. However, according to our study, most are designed to support URL addressing with Chinese characters in between. Few if any works on the e-mail addressing issue. This paper presents a description of our experimental system, which supports localized Chinese e-mail address mapping by using the LDAP directory service. In the future, if there is any standardized multilingual addressing scheme available, it could be incorporated into our system. The same user interface could still be used. With minor modifications, the same approach should be easily adapted for utilization in other language system.

Keywords: **Chinese, LDAP, URL, web-mail**

1 Introduction

E-mail is one of the most popular services on Internet. Fast message transportation, good GUI designs and enhanced localization capabilities in end-user environments are the primary keys. Most people would like to communicate with each other via their native language(s) if possible. By the efforts of computer scientists, most of us could write and read e-mail contents in local language today. However, as to the addressing part, that is another story. There still exist some addressing problems in internetworking for many users because they are based on the 7-bit ASCII character set. For those who do not know English well, ASCII character set is hard to memorize and is prone to making errors. With more users joining the Internet, this kind of addressing problems should not be ignored.

Up till now, there is no multilingual addressing standard, no multilingual registry in gTLD[11], ccTLD on the DNS[7] naming infrastructure. Currently, RFC 1035[7] is the main implementation obstacle. It limits the valid domain name character set to be a subset of the ASCII character set. Furthermore, while using in DNS, all the capital letters and their corresponding little characters are treated as the same by historical reason. These make the non-ASCII addressing still not possible in general.

There are many proposed approaches [10][13][17] to support the Chinese and/or multilingual DNS name resolution. To name a few, internetworking scientists in Asia Pacific region (including China, Hongkong, Japan, Korea, Singapore, Taiwan, etc.), RIPE, etc., are undertaking some IDN projects for developing multilingual addressing environments. However, according to our study, most of the proposed solutions are mainly designed to support the URL addressing with multilingual characters in between. Few if any addresses the e-mail addressing issue.

We had designed and implemented a web-mail system with Chinese addressing capabilities by incorporating the LDAP directory services [3][4]. Conceptually, we could view an ASCII e-mail address as one of the attributes of some user's profile. By storing users' profiles on directory servers, software with directory-enabled capabilities could be easily used to extract the ASCII e-mail information for further utilization. In this way, our system provides a workaround solution for the Chinese e-mail addressing problem indirectly by translating a Chinese name to its corresponding ASCII e-mail address. The same approach could be easily adapted for utilization in other language system.

1.1 Chinese E-mail addressing

Every e-mail message could be divided into two parts: the header and the body. Now the problem to send messages with multilingual characters in the body can be dealt with by using MIME[1]. Before sending, the sender programs encode messages with the MIME standards. The messages are then transported over the Internet to the destinations. At last, they are decoded with MIME enabled clients. However, as for header sections, we still do not have a standard solution for non-ASCII addressing. Up to now, almost all mail client and server software on Internet communicate with the ASCII addressing expression only. Could there be any systematic approach (or workaround solutions) for supporting non-ASCII e-mail addresses?

To further describe the main ideas, let us check the three (pseudo) e-mail expressions shown below:

- jsc.cis84@nctu.edu.tw
- Jian-Shyong Chen< jsc.cis84@nctu.edu.tw >
- • • • •< jsc.cis84@nctu.edu.tw >

On the first look, they seem different; however, basically they mean the same thing. That is, jsc.cis84@nctu.eu.tw is the true component for e-mail address routing. However, using LDAP addressing book, the last form with Chinese name could be a good candidate for providing a workaround solution to support multilingual addressing.

1.2 Web-based Environment

WWW browsers have become standard applications for Internet access today. For those who do not have their own computers on working places (e.g. students in school environments), there are some obvious advantages through using browsers to send or receive mails:

- No additional software is necessary. All one need is a browser program.
- The browser programs (e.g. IE, Netscape, etc.), being the most popular software, can often reduce the learning time of users.
- Web-mail systems could be easily adapted for supporting roaming access.

1.3 Mail Routing with Directory Service Support

By putting e-mail addresses with forms like the last one shown above onto LDAP directory servers and through the translation of directory-enabled web-mail systems, we could achieve the goal of communicate with multilingual addressing indirectly. This is promising for many people.

Let us describe the working paradigm shown in Figure1. The mail routing is performed as below.

- The user types the Chinese name of the recipient (for example, "• • • "), together with additional information (e.g. school name, city name, etc.), through the client mail interface.
- The client then consults the LDAP server to see if there is any one matching the search condition. If yes, all the matched people's information will be returned to the client, and the user can choose the one wanted. After that, the LDAP Server will return the related ASCII e-mail address of the recipient
- Using the returned ASCII e-mail address, the client sends the message to the destination SMTP server.

In principal, there is no need to alter the original mail server routing. It works as before on one condition. That is, if there are distributed LDAP servers on the related sites, with chaining and referral capabilities enabled.

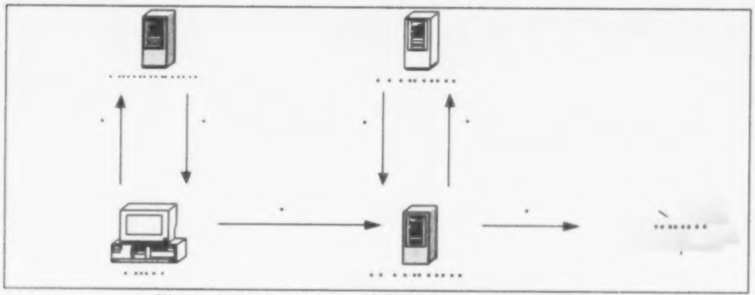


Figure 1. Mail routing with directory server support

2 Related Work

There are several proposed approaches to solve the Chinese URL addressing problem. Three of them will be described briefly in Sec.2.1, Sec.2.2, and Sec.2.3. Interested users are encouraged to visit the related web sites for more details. In Sec.2.4, we will describe the main practical problems of these systems.

2.1 mDNS [13]

The mDNS project is under joint development by researchers in TWNIC, Academia Sinica, and National Central University, Taiwan. The goal is to develop an internationalized DNS system to help the "non-English" DNS architecture to become standardized. mDNS would not effect the existing gTLD or ccTLD. Although the proposed experimental architecture can accept Chinese (BIG5) URL, it is not full Chinese URL. It is necessary to modify the source code of the ".tw" root server; however, it is not necessary to modify the existing client software.

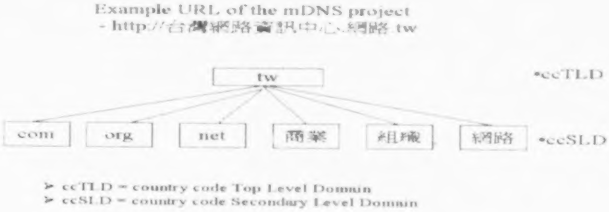

Example URL of the mDNS project
- http://台灣網路資訊中心.網路.tw
tw
•ccTLD
com
org
net
商業
組織
網路
•ccSLD
➢ ccTLD = country code Top Level Domain
➢ ccSLD = country code Secondary Level Domain

Figure 2. The experimental architecture of the ccTLD ".tw"

2.2 iDNS [17]

The iDNS project is mainly under development by Singapore researchers. As shown in Figure 3, the key component of iDNS is the domain name proxy server, which translates the i18n (internationalization) domain name to the format of UTF-5 and transmits the translated format to the real DNS server. When the iDNS system receives the ASCII domain name, it will consult the old existing DNS system. However, if it receives the non-ASCII domain name, it will be routed to the i18n branch system.

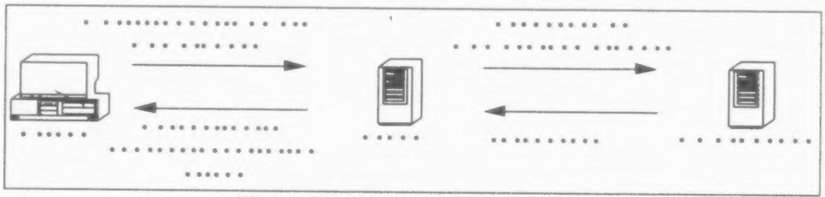


Figure 3 iDNS domain name proxy server

2.3 cDNS[10]

The cDNS project, run by researchers in CNNIC, is developing similar scheme. The main idea of cDNS is the proposed ***DNS forest*** architecture instead of the traditional rooted DNS tree structure. Interested users are supposed to refer to the web site of CNNIC for further details.

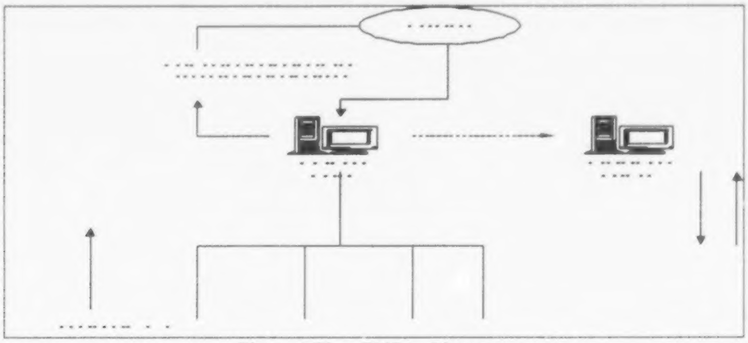


Figure 4 The cDNS architecture.

2.4 Practical Application Problems

Although it looks promising in the first place; however, up to now most of the proxy/caching and mail servers (ex. Squid and sendmail) cannot accept non-ASCII addressing. As mentioned in the introduction section, RFC 1035 is the current implementation obstacle, which breaks all the paradigms. It's nearly impossible to keep compatibility with the current system without modifying the source code of these servers, recompiling and reinstall the systems.

3 Overview of the LDAP Directory Service

It seems that we could not get an immediate multilingual addressing solution without modifying the existing servers all over the Internet. That is why we think that the LDAP enabled web-mail system might be a good workaround solution to try. Before further on, let us make some introduction on LDAP.

3.1 Why LDAP?

Historically, X.500 [3][4] is based on the ISO stack. It is just too complex and hard to implement in the real environments. LDAP [3][4][8][9] is the protocol initially designed as one front end of X.500. Since LDAP can be easily implemented and can be used to exchange information between LDAP servers, standalone LDAP directory server becomes popular gradually. As shown in Figure 5, Desktop PC can access the data of LDAP/X.500 directory server by LDAP protocol.

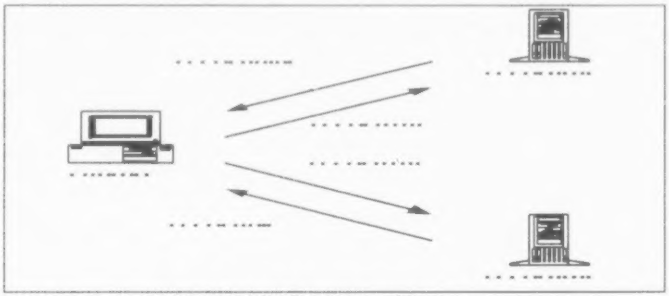


Figure 5 LDAP system architecture.

3.2 How Does LDAP Work?

LDAP stores the information hierarchically, where data is stored as key-value pair. [4][6] Each key will be mapped to one or many values. For example, *cn* (common name) will be used to store user name, mail will be used to store user's email address, etc [2][4]. Every node in the tree architecture of the LDAP directory will be viewed as an object, which has one or many *objectclass* attributes to identify the node. The *objectclass* definition of the system is that it should have what kind of attributes and it is allowed to have what kind of attributes. We also can extend the original *objectclass* by adding the attributes we want. Every node in the tree will be identified by *DN* (Distinguished Name) attribute [4][6]. The hierarchical relationship of the tree can be divided by locality or corresponding categorization. For example, the directory tree could be constructed according to the domain name. From the top level, c=tw (country code), o=edu (organization), ou=NCTU (organization unit) to ou =CIS (organization unit), the information in an example leaf node is a student named jacky, studying in the Dept. of Computer and Information Science. Through the tree architecture, LDAP clients can perform search, delete, modify operations and any site manager can add ACL (Access Control List) mechanism to control the access. [3][4] For more details, interested users are encouraged to read the related LDAP documents listed in the reference sections. [3][4][6][15]

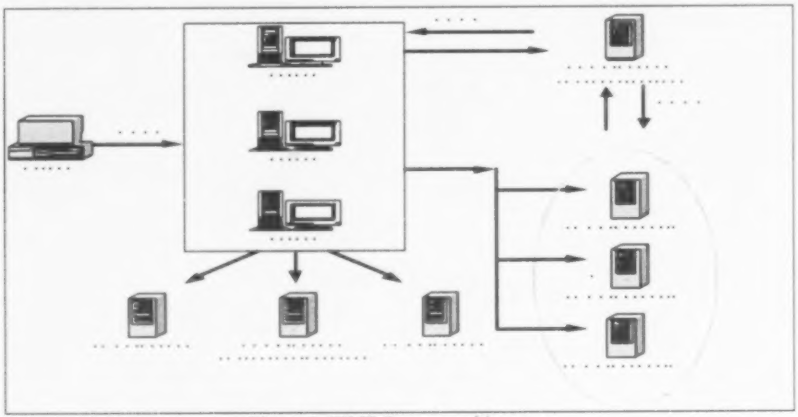


Figure 6. EIMP System architecture

4 Our System Design

We built up an EIMP (Enhance IMP; based on the IMP[12] project) system on a Linux platform [14][15]. The main system components are Apache+PHP(WWW), IMAP+Sendmail(Mail), OpenLDAP(LDAP), etc. We enhance the web-mail system by integrating the LDAP directory service and adding the Chinese naming capabilities. As shown in Figure 6, users can connect the logon server through www browsers and types his/her user name (Chinese or ASCII) and password. After passing the authentication, the LDAP server will return the user's IMAP[14] server location to the logon server. Then it will connect the related IMAP server to retrieve the user's mail(s). On the other hands, users can send mails through the SMTP server and store their address book information in the database server. Thus, for users wishing to send mail to his friends, even if they do not remember the exact email address, they can still find the email addresses by the help of LDAP servers, as shown in Figure 7.

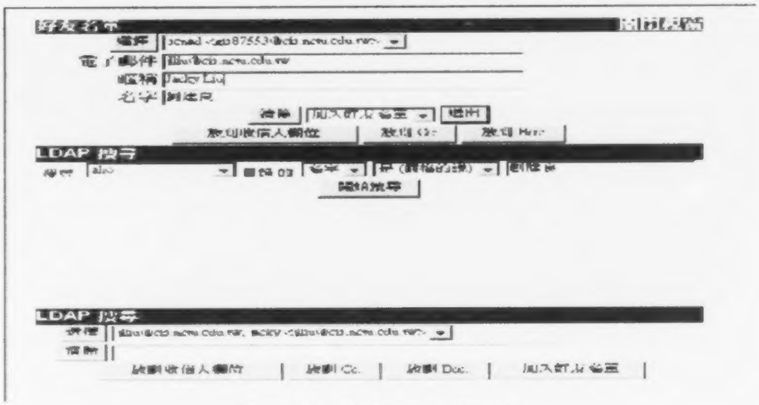


Figure 7. LDAP search screenshot

5 Problems and Discussions

As we know that, although it looks promising; however, up to now most of the proxy and mail servers (e.g. Squid and Sendmail, etc.) cannot accept non-ASCII addressing. Hence, it's nearly impossible to keep compatibility with the current system without modifying the source code of these servers, recompiling and reinstall the systems.

Non-ASCII communication issues are new hot topics in many research applications. While this is true in the DNS system, similar situation appears on the LDAP protocol suites. As more directory servers have been set up, there are more chances for directory servers to share and exchange their information through chaining or referral [3][4][5]. LDAPv3 addresses the issue by using the UTF-8 [16] encoding, while LDAPv2 use T.61, which lacks the capabilities to handle 8-bit data. Hence, in building distributed Directory server groups, software packages with LDAPv3 ready should be the proper choices.

We choose the IMAP (and not POP3) protocol for serving our mail access since IMAP servers support both the offline and the online modes. Currently, most sites use POP3 as the access protocol since it is simple and usually gives low impact to the system performance. However, as the POP3 protocol operates only in off-line mode, it could not meet the new trend for supporting the e-mail roaming access.

6 Conclusions

E-mail is one of the most used Internet applications today. However, non-ASCII addressing system is still a research issue. Most of the proxy and mail servers (e.g. Squid and Sendmail, etc.) still cannot accept non-ASCII addressing. Internetworking scientists from members at APNIC, RIPE, etc. are undertaking some piloting projects for producing multilingual internetworking standards. However, no one knows when the solutions will be ready. This is not good for primary/secondary school education on most parts of the world.

In this paper, we describe our approach for the Chinese e-mail addressing and authentication problems. We have designed and implemented a web-mail system with Chinese addressing capabilities by incorporating the LDAP directory services. Conceptually, we could view an e-mail address as one of the attributes of some user's profile. By storing users' profiles on directory servers, software with directory-enabled capabilities could be easily used to extract the ASCII E-mail information for further utilization. In this way, our system provides a workaround solution for the Chinese e-mail address problem by translating a Chinese name to its corresponding ASCII e-mail address. This should be promising for many people. For example, using such LDAP-enabled web-mail system, primary/secondary school education could benefit a lot since most of the communication activities could be done in their local native language.

In the future, if there is any standardized multilingual addressing scheme available, it could be incorporated into our system. The same user interface could still be used as well. Furthermore, with minor modifications, the same approach should be easily adapted for utilization in other language system.

Acknowledgment

This work was supported by the National Science Council of the Republic of China under Grant No. NSC89-2511-S-009-014. We would like to thank the anonymous reviewers for suggesting improvements and offering encouragement.

References

[1] Freed, N., Borenstein, N. "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, (1996)
[2] Howes, T., etc., "The String Representation of Standard Attribute Syntaxes", RFC 1778, (1995)
[3] Howes, T., etc., "A Scalable, Deployable, Directory Service Framework for the Internet", INET 95
[4] Howes, T., etc., "Understanding and Deploying LDAP Directory Services", Macmillan Technical Publishing, (1999)
[5] José, Rui J.P.; Costa, António; Macedo, Joaquim; Freitas, Vasco, "Providing multiple external views on directory user interfaces" in Computer Networks and ISDN Systems, Vol 28, No. 4, pp. 543-550, (1996)
[6] Kille, S., "A String Representation of Distinguished Names", RFC 1779, (1995)
[7] Mockapetris , P., "Domain Names--Implementation and Specification" ,RFC 1035, (1987)
[8] Wahl, M., etc., "Lightweight Directory Access Protocol (v3)", RFC 2251, (1997)
[9] Yeong,W., Howes, T., Kille, S., "Lightweight Directory Access Protocol", RFC 1777, (1995)
[10] Chinese Domain Name Consortium, http://www.cdns.org/e-index.shtml (accessed on August 1,2000)
[11] ICANN, http://www.icann.org (accessed on July 1,2000)
[12] IMP@horde.org, http://www.horde.org/imp (accessed on April 23,2000)
[13] mDNS TANet Testing Project, http://mdns.twnic.net.tw, (accessed on June 1,2000)
[14] Message Access Paradigms and Protocols , http://www.imap.org/papers/imap.vs.pop.html (accessed on May 9,2000)
[15] LDAP Linux HOWTO, http://www.linux.org/help/ldp/howto/LDAP-HOWTO.html (accessed on May 10,2000)
[16] Unicode - Basic Questions, http://www.unicode.org/unicode/faq/ (accessed on June 1,2000)
[17] Welcome to iDNS , http://www.idns.org (accessed on June 1,2000)

Design and Implementation of A N-Tiered Heterogeneous Virtual School Administration System

Huang Gooshyon
Tamkang University Information Center
151 Ying-chuan Road, Tamshui, Taiwan 251
Tel: +886-2-2621-5656-2681
E-mail:georgehw@mail.tku.edu.tw

There are two types virtual school administration systems, web-based or voice-based, which are currently used by students. They are systems with different access mechanisms but same business logic, and require two times of resources for development and maintenance. Whenever the business logic of the systems changes, both of the systems need to be implemented. As the wireless communication grows more popular, the school has been considering adding a wireless interface to the system. However, with current architecture, the only way to add a wireless application protocol (WAP)-based system is to implement an additional system from scratch. Since the voice-based system and the web-based system have the same business logic, they can be integrated into one. We can dedicate an application server for the business logic, which interacts with the web-based interface and the voice-activated interface with a set of application programming interface (API). With the extraction of the business logic and the business logic API, developers for the voice-activated interface and the web-based interface can implement the interfaces without specific knowledge of the business logic of the system. With this design and architecture, the system can be further expanded to support a WAP-based interface and other interfaces easily.

Keywords: **Internet, wireless, virtual school, heterogeneous**

1 Introduction

The Internet is widely used for school education, especially virtual school education [2][3][4]. The advantage of the Internet is its capability of supporting multimedia and its attractiveness to the user. For the virtual school education, the students study via the Internet. They do not have to be in the classrooms of a school and can learn at anywhere at anytime. However computers and communication networks are needed to support virtual education through the Internet. The cost of the computers and setting up the communication networks is very expensive. Thus, the systems are not available everywhere. Furthermore, system interfaces must be developed in order to allow the users to access the computers and the networks. The purpose of these system interfaces is to provide an easier way for the students to access the systems and to allow the students to interact with the instructors real-time. Those systems interface do not need to be attractive and colorful since its main goal is to provide a mechanism for the students to access information real-time. For a web-based system, the homepage can be design in a way to reduce the network traffic and system load. However, not every student can access the computers and the networks due to his financial situation or the load of the system. For the students who cannot access the computers and the networks, the telephone (the voice activated based interface) provides another popular access media. Therefore there are needs for systems to support both telephone (voice- based) and web browser (web-based) interfaces [1]. The web-based system is more visual and more user friendly, however, the voice-based system is more convenient, more affordable, and requires no hardware investment from the students [illegible] the technology evolves, the wireless communication is gradually taking over the traditional wire line communication. To support the wireless communication the system will need to be expanded to support the wireless application protocol (WAP)-based interface [10].

Originally, a couple of the school administration systems we had can be accessed via a regular telephone or via a web browser but not both. They were basically two different systems, though they support the same business logic. Both of them have their own user interface and system logic and were designed, implemented, and maintained separately. To support them two sets of resources are needed. The original system architecture is shown in Figure 1. Developers for both of the systems handle both the business logic's and the user interface's design and implementation. Whenever the business rule changes, both of the systems need to be modified and updated. It is very costly and difficult to keep both of the system consistent.

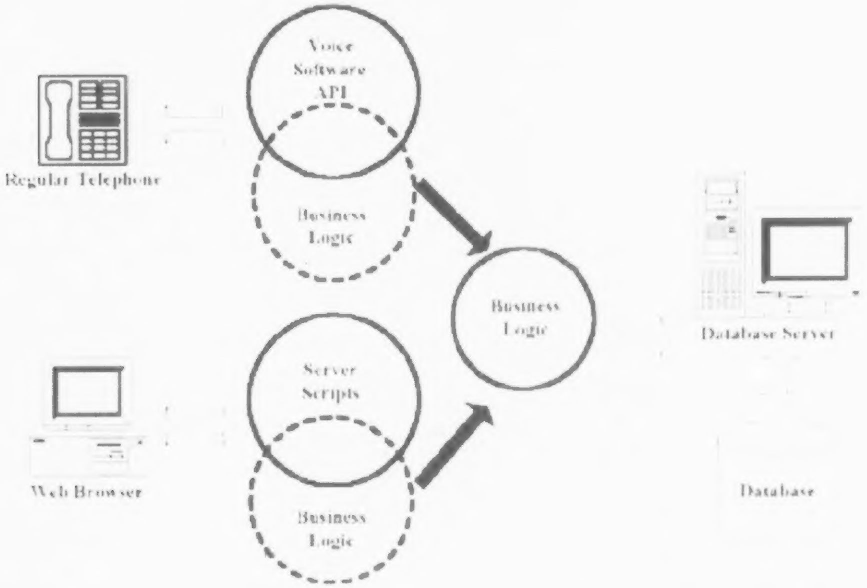


Figure 1. Logic view of voice system and web system

To reduce the maintenance cost of the two systems and to make them easier to be upgraded and expanded, we have proposed to integrate the two systems by extracting the business logic module out of them and migrate it into an application server. The remaining of the systems is migrated into a web server and a voice server respectively. By doing this, we dramatically reduced the cost of maintaining the system. After the architecture change, whenever there is a business change, only the application server is affected. We reduced the maintenance cost by 50%. No more concerns about the consistency of the systems. With the modification of the system architecture, we make it more scalable and expandable. The system can be easily expanded to support other access media without making changes to the application server. For example, to support a WAP-based interface, a WAP server can be easily introduced and integrated into the modified system architecture.

2 System Architecture and Implementation

2.1 Architecture

The administration system is an N-tiered system.

- Data Services Tier: The database services and implementations.
- Business Logic Tier: The business rule of the system.
- Translation Tier: Translate the I/O between application server and gateway server. For the voice-based system, the gateway server is the voice server. The purpose of the voice server is to translate PSTN and HTTP between application server and usual telephone. For the web-based system, the translation tier is transparent; it does not do anything. For the WAP-based system, the WAP Gateway is the gateway server. The purpose of WAP Gateway is to translate the WSP/WTP and HTTP between WAP telephone and web server.

- Presentation (UI) Tier: The input and output of the web-based system is HTML. The input and output of the voice system is the key press and voice of usual telephone. The input and output of the WAP-based system is WML [10].

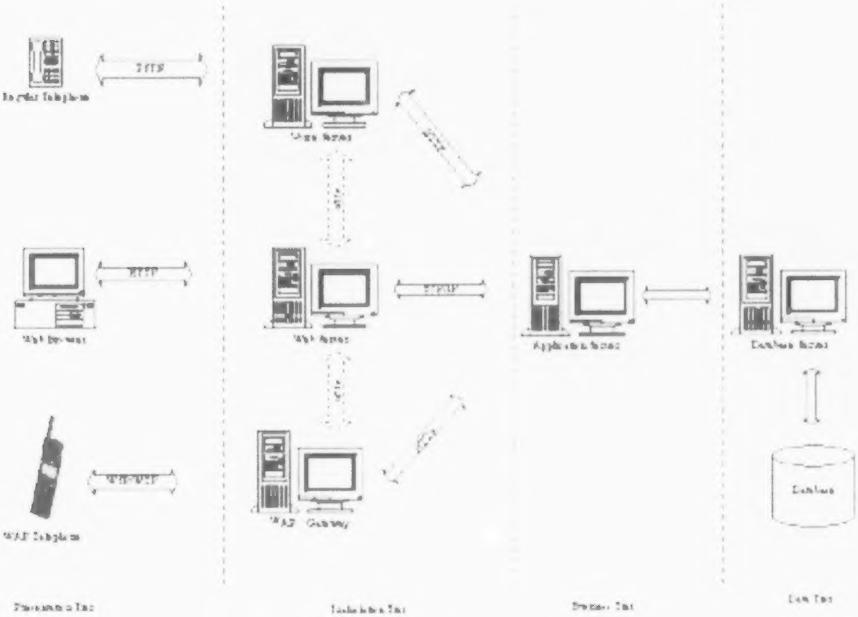


Figure 2. System Architecture

In the Architecture, the application server is the most important part. The application server needs to process business logic and interact with voice server, web server, and WAP Gateway. Because the protocol between the application server and the voice server and the WAP Gateway is HTTP protocol, we can set the application server and the web server in the same machine. The developers of the application server are more responsible, because they must handle business rule, HTML and WML. The developers of the other systems implement User Interface and do not have the knowledge of business rule of the system, because the developers of the application server handle the business rule. The application server sends different output format to different systems by parameters. Under the Architecture, after building the web system, the other systems are easily to build.

2.2 Architecture of the Voice System

Because taking business logic out of the voice system, the function of voice system is coherent. It translates the output of web server to telephone. The output format of web server is HTML. So the voice server has to simulate to web browser, shown as in Figure 3.

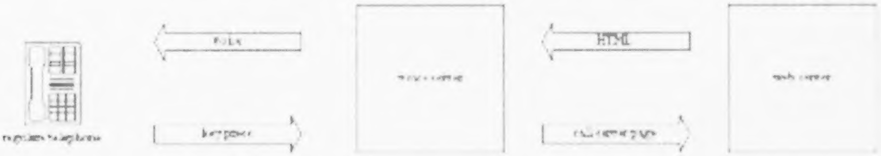


Figure 3. The communication methods are between voice server and web server

3 Case Study

The Enrollment System of the Tamkang University [7] is designed and implemented following the architecture of this paper, shown as in figure 4. The system has been deployed and used by thousands of concurrent users [8].

3.1 Hardware Structure

We used thirteen Pentium based servers to implement the system. Six of them are used as the web servers. One machine is used as the UNIX Gateway. One server is used as the alert and automating email server. Four voice servers are used to support the voice activation. Finally, all student enrolment information is stored in one database server. The network hardware are two 100 MB/sec switch hub.

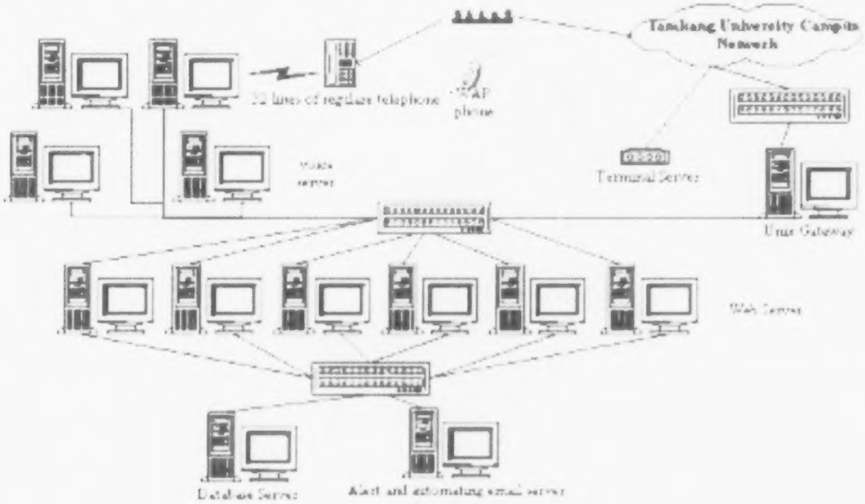

Tamkang University Campus Network
32 lines of regular telephone
WAP phone
voice server
Terminal Server
Unix Gateway
Web Server
Database Server
Alert and automating email server

Figure 4. System Hardware Structure

3.2 System Software

OS: Microsoft NT4.0 is used for the web servers, voice servers, and the alert and automating email server. Free BSD 3.0 is used for the UNIX Gateway [8].
Web server: Microsoft IIS 4.0.
Database: Microsoft SQL Server 6.5.

3.3 Load Balancing and Scalability

To make the system suitable for all schools, we also took into considerations of the cost of hardware and the scalability of the system. A set of low-end servers can be grouped together to replace a high-end server[6]. To achieve this, a DNS server is needed for the load balancing work. The simple round robin methodology is used for the load balancing. With the current flexible four-tiered architecture, servers can be added into the system to share the performance load whenever the system load is heavy[9].

3.4. Security

Two security strategies are used to increase security:

1. Packet filter: It only allows IP packets through port 80 to access the web server, the packets of the other ports can not pass through. The web system can avoid being attacked by the other machines.
2. Supports multi-protocol: TCP/IP protocol is used between the web server and outside systems. IPX protocol is used between the web server and the database server. The web server should be hacked, the database server is kept away Internet and the database is still safe.

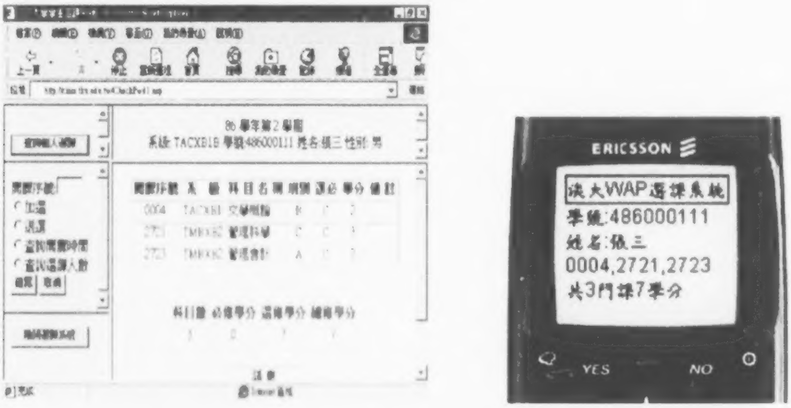
3.5. Network Management and Monitoring

The alert system has the following features:

1. Monitoring the system: It sends to keep-alive message to web servers, voice servers, and database servers in every period.
2. Network management system: Checks network traffic between web servers, voice servers and database server.
3. Auto Backup the data of database server.

3.6. User Interface Design

One of the most important criteria of the virtual school administration system is to let students access and retrieve correct information real-time. The user interface must be simple to reduce network traffic and system download time. The homepages for the web system and WAP are simple and straightforward to improve system performance. The look and feel of the WAP homepage depends on the WAP telephone the user users. An Ericsson r320 model WAP homepage is shown here as a sample WAP homepage. We can compare the home pages for the web system and WAP system.



淡大WAP選課系統
學號:486000111
姓名:張三
0004,2721,2723
共3門課7學分

Figure 5. The display of the homepage of WAP-based system

3.7 Log statistics and analysis

Duration of enrollment period, the system generates the log automatically everyday for statistics and analysis, as shown in Table 1.

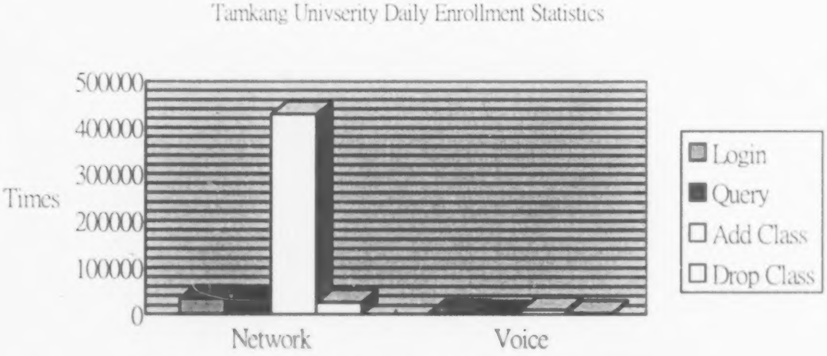


Table 1. Tamkang University Daily Enrollment Statistics

By comparison, the load of the web system is much heavier than the load of voice system. Since the voice system has 32 telephone lines, it can only support 32 concurrent users. In the peak hour of the enrollment (the first hour of each grade enrollment), the load of the web server is high.

We expect the voice system and the WAP system to be fully loaded during the peak hour. A dedicated business logic-processing server is used for the voice system and the WAP system. Since the telephone lines of the voice system and the WAP system are limited (up to 32 lines), a dedicated web server for the business logic processing of the voice system and the WAP system is sufficient.

4 Conclusions and future development

The development and maintenance resource of the heterogeneous systems depends on how many access media. The more access media, the more resource it needs. My proposal has the following advantages:

- Resource Reducing: Because the business logic is centered, heterogeneous systems need one business-logic process only, the resource of development and maintenance is less than usual systems.
- Expandability: With the N-tiered system architecture design, the business logic system was designed and implemented to support different UI systems. Different UI access method can be easily added into the system.

In the system, the application server interacts with voice server and WAP Gateway on HTTP protocol., so the application server must have functions of the web server. We can develop a new structure of the application server for voice-based system and WAP-based system, and the application server interacts with the voice server and WAP Gateway on TCP/IP.

References

[1] C. Chou and C. T. Sun, "A Computer Network-Supported Cooperative Distance Learning System for Professional Communication Education", IEEE Transactions on Professional Communication, Vol. 39, No. 4, pp. 205-214, (1996).

[2] Colin McCormack & David Jones, "Building a Web-Based Education system", John Wiley & Sons, (1998).

[3] Dwyer, D., Barbieri, K.& Doerr, H. M., "Creating virtual classroom for interactive education on the Web", Computer Networks and ISDN Systems 27, pp. 897-904, (1995).

[4] Nian-Shing Chen, "Web-based Learning Environment", Information and Education, Vol. 64, April, pp. 2-13, (1998).

[5] C. T. Sun, C. Chou & B.K. Lin, "Structural and Navigational Analysis of Hypermedia Courseware", IEEE Transactions on Education, Vol. 41, No. 4, CD-ROM, November, (1998).

[6] Chen-Chung Liu, Gwo-Dong Chen, Kuo-Liang Ou, Baw-Jhiune Liu & Jorng-Tzong Horng, "Managing Activity Dynamics of Web Based Collaborative Applications", International Journal of Artificial Intelligence Tools, Vol. 8, No. 2, (1999).

[7] Gooshyon Hwang & Chao-Tsong Fangtsou, "Introduction of Tamkang University Enrollment System", Ministry of Education Computer Center Newsletter, Vol.8802, (1999)

[8] Jack Chou, "NT Dynamic Web Server Design", Unalis Company, (1997)

[9] J.J.P. Tsai, S.J. Yang & A. Liu, "An Integrated Architecture for Distributed Real-Time Systems Analysis", in IEEE Second World Conference on Integrated Design and Process Technology, Austin TX., pp. 152-159, (1996) Dec. 1996

[10] WAP Forum, http://www.wapforum.org/what/whitepapers.htm (accessed on July 1, 2000)

Design and Implementation of a WWW - Based School Official Memorandum System

Gi-Ping Lee, Yue-Shan Chang and Ching-Chuan Chou
Department of Electronic Engineering, Ming-Hsin Institute of Technology
1 Hsin Hsing Road, Hsin-Fong, Hsin-Chu, Taiwan 304 R.O.C.
louis@mheman.et.mhit.edu.tw, ysc@mhit.edu.tw, ruth@mhit.edu.tw

1 Introduction

The official memorandum is a very important document that offers a decision path of something in the most organizations. In general, the executed policy usually needs agreement of decision-maker through official memorandum. All the official memorandums are traditionally passed by one-by-one human delivery from faculties to managers in an organization. It may results in the lower performance of administrative. Even though some administrative operations are via e-mail or other approach. It have some limitations, such as the official documents usually need the signature of decision-maker, it is not ease to overcome via the e-mail.

In order to have a speedy the administrative operation environment, especially the official memorandum delivery, we design and implement a WWW-based Official Memorandum System in a school. That is a WWW application without any novel theory and technique. We apply the existing techniques used in the WWW environment to accomplish the application.

Obviously, the system is based on the client-server model. Implementing the system has some existing techniques can be used, such as CGI, Java/Servlet [3], Java/CORBA [3]. Lotus' InterNotes [1] product uses CGI mechanisms to allow Web browser access to documents and forms managed by the Notes Server. Documents to be placed on the Web are translated by a program to HTML. These documents and forms are accessed through a standard HTTP server as though they were normal HTML documents. Java is a portable object-oriented language, and also a good platform for writing client/server web-based applications. Servlets are secure protocol and platform-independent server side web-enabled software components, written in Java. Java/CORBA has a clear advantage over CGI solution, such as flexibility, maintainability, and responsiveness etc.

Security issue in the system will be taken care by using traditional approaches. There are two secure mechanisms will be used: one is account/password, the other is the firewall. First one can prevent non-authority user log-in into system and disrupt the system. All the general users must apply for an account excepting the chief of department. And the system will force all users to change the password periodically. This mechanism can avoid internal hackers. Second one is to avoid external hackers who intrude into system for non-authority accessing. Few hackers, of course, can intrude into and disrupt the system. Some approaches can be used for enhancing the security of information, such as data compression/decompression before accessing to/from database and checking the data consistency of duplicated database periodically. All of them are the future works

Fault tolerance is in order to enhance the reliability of system. In fault-tolerance community, many approaches have been proposed to enhance the data reliability [4,5]. The approach in the system is database replication. We use warm stand-by primary/backup scheme to improve the system availability. Many issues in the data replication that have to be guaranteed are employed like the [5]. These issues are such as idempotent operation, data consistency, and recovery. Because the system is a three-tier scheme, all operations supporting fault-tolerance are implemented in the core of the system. This feature can also prevent the database crash during the formal execution phase.

A complex system has to be manageable in an easy way. In order to enhance the system flexibility, a web-based management tools should be implemented. System manager can add and remove user easily. In addition, system manager can also maintain the database, such as record manipulation, in an easy way.

Many features are described previously. In addition, we will support some important functions shown as following: Official documents writing, Official documents progression tracking, Auto-delivery, Automatic signing, Urgent document notification.

2 Design and Implementation

According to the described above, we design the system architecture like as Figure 1. The architecture is simple and complete. The system includes an Official Memorandum System and a replicated database. The system will receive requests from clients. For security issue, we add a firewall in the front of web server. All the requests must be checked by the firewall for ensuring the request is an authority request. In addition, the Official Memorandum System is responsible for all the features described above, which include fault-tolerance. A replicated database is also included in the system. The database used in the system is the SQL database.

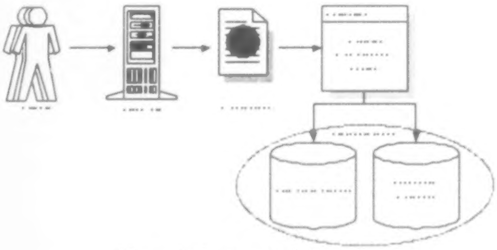


Figure 1. System Architecture

The whole system is implemented and run on the Windows NT 4.0 and SQL server 7.0. The programming paradigm is ASP that using VBscript. With the fault-tolerant, the system needs to access primary and standby database separately. To guarantee the consistency of two databases, we apply the traditional two-phase commit protocol on the replicated database transaction processing.

Figure 2 shows the GUI of document reviewing for those chiefs of department. When they login into the system, the system will show the urgent document on top of the reviewing page, which indicate these documents have to review first. The document reviewing process will sign the signature automatically when the process achieved.

Figure 3. The GUI of Document Reviewing

3 Conclusions

In this paper, we have been stated the design and implementation of a web-based official memorandum system. This system can migrate the conventional official memorandum system to network. That is a WWW application without any novel theory and technique. We apply the existing techniques used in the WWW environment to accomplish the application. In order to avoid the informal accessing to this system, the firewall is utilized at the front-end of the system. Besides, the duplicated databases are used in this system to prevent the databa[illegible] rash during the formal execution phase.

References

[1] Lotus Development Corp., "Lotus InterNotes Web Publisher", Setember 1995, http://www.lotus.com/corpcomm/334a.htm.

[2] Evans and D. Rogers, "Using Java Applets and CORBA for Multi-Users Distributed Applications", IEEE Internet Computing, Vol. 1, May-June 1997, pp.43-55.

[3] Guang-Way Sheu, Yue-Shan Chang, Deron Liang, Shyan-Ming Yuan, Wintson Lo, "A Fault-Tolerant object service on CORBA" in Proc. of the 17th International Conference on Distributed Computer Systems (ICDCS'97) Baltimore, USA, pp.393-400.

[4] Yue-Shan Chang, Jyh-Tzong Chiou, Shang-Rong Tsai, "Highly Reliable File Services on the DMINIX System, " The 11th Technological and Vocational Education Conference of R.O.C., Vol.3, Mar. 1996, pp. 30-39.

Design and Implementation of Teaching Models in Web-Based Teacher Training

See-Min Kim*, Shin-Chon Kang, Young-Kyun Back****
** CRADLE, Tokyo Institute of Technology*
2-12-1, Ookayama, Meguro-ku, Tokyo, Japan 152-8552
Tel: +81-3-5734-3233, Fax: +81-3-5734-2995,
E-mail: kim@ak.cradle.titech.ac.jp
*** Korea National Universty of Education*
Darak-ri, Kangne-myon, Chongwon-kun, Chugbuk, Korea 363-791
Tel: +82-431-230-3431, Fax: +82-431-233-0664,
E-mail: skytop@chollian.net

The reform of teacher training started in May 1995 in the Republic of Korea with reform of the educational system. The core of the reform was reinforcement of teacher training activity and introduction of a DTTS(Distance Teacher Training System). Then, in order to introduce a DTTS, the project for distance teacher training model development started in September 1997. This paper is related to a design and implementation of a teacher model in a DTTS. The teaching models of the following 4 types were carried out. 1) Problem-Solving type, 2) Seminar type, 3) Lecture-Practice type, 4) Courseware type. This system was in operation from October 1998. Current problems of this teacher models include: 1) Poorness of course contents, 2) The difficulty of checking a learning process, 3) Insufficiency of feedback to a trainee etc.

Keywords: **Distance Teacher Training, Teacher Model, Web-Based Learning**

1 Introduction

In Korea, reform of teacher training started on May 31, 1995 with the announcement of a reform of the educational system proposal. Philosophical bases for reform of teacher training are the spirit of the opened education, enshrining the principles of, opened an educational opportunity, the learning speed, the contents of learning, and the learning method, etc. The contents of reform are as follows[5]:

1) Obligation of periodical training;
2) Execution of distance education that introduced high technology of information communication engineering;
3) Reflection to the personnel affairs and the salary of a training result;
4) Authorization of a special course completion result in a graduate school and a social-education organization;
5) Attempt to the improvement of the training organization that enabled selection of the training organization by the teacher and let competition pass in qualitative.

These are summarized the following: 1) reinforcement of teacher training activity 2) introduction of a DTTS(distance teacher training system). It aimed at an expansion of the training opportunity, and overcoming restriction of time and space, with a reduction of training expenses. The project of DTTS development started in September 1997. It was sponsored by the Korea Multimedia Education Center. This project was divided into 4 sub-projects: Develop a training support model, design for teaching model, courseware development, and development for system management model. This paper is related to a design and implementation of a teacher model in a DTTS.

A teacher model is dependent on the contents of course, the learner characteristic, learning environment, etc [6]. According to the questionnaire for the teachers and educational professionals of Choi [2], the suitable course for distance teacher training is as follows.

1) Various culture subjects (humanities a subject and a theoretical field).
2) Teaching methods expected such as discussion and workshop, then a lecture.

In Korea, as a training course into the distance teacher training, the culture subject of 11 was chosen. These were, "Foreknowledge of the future society and a counter plan", "Understanding of traditional culture", "The world in the 21st century and the Korea", "An information society and a computer", "Environment and education", " Raising of national morality nature", "An information society and multimedia education", "Theory and practice of open education", "The direction of the educational system reform and school reform", "Education of humanity and originality", "Education for a unification counter plan".

In consideration of the characteristic of subjects and learner, strategies of WBI(Web-Based Instruction)[1], the teaching model of the following four types was proposed. 1) Problem-solving type, 2) Seminar type, 3) Lecture-practice type, 4) Courseware type. These are described at length in sections 2-4.

2 Design of the Teaching Model

In this project, the model of distance teacher training was divided into the macro model and the micro model, and was developed accordingly. A macro model is the framework of the whole DTTS, and a micro model is the course of training, that is, a teaching model. A macro model and a micro model are unified and distance teacher training is managed.

2.1 Web-Based Instructional Strategies

The acquisition process of the knowledge in WBI and the approach of the learning of constructivism are very similar. The most basic principles of constructivism concern fundamental philosophical assumptions about knowledge and learning[4]. The first, more generally accepted principle is that what a person "know"is not passively received, but actively assembled by the learner. The second principle is that learning serves an adaptive function. That is, learning is not the storage of "truths,"but of useful personal knowledge. This means the importance of the context of learning. Context has a lot to do with what is perceived as useful knowledge and how what is learned is integrated with existing knowledge. And the assumption that education is about acquiring universal truths. Since each person has different experiences and constructs an individual account of these experiences, each persons' s reality is slightly differenbt. New expereriences are interpreted within the context of these individual realities, implying that each person "know"a particular thing in a slightly different way.

We introduced the application of a repertoire of cognitively oriented instructional strategies implemented within a constructivist and collaborative learning environment, utilizing the attributes and resources of the internet[1]. The instructional strategies may be designed the following ways:

1) Support to the interaction between a lecturer- learner, and a learner-learner.
2) Introduce a hyper-textual function and support individualization learning.
3) Various learning materials provide in real time or non-real time (multimedia support)
4) The contents of learned and an evaluation results are analyzed rapidly and correctly, and it offers feedback to learner and system side.
5) Provide of DB Retrieval Function for learning information
6) It cooperates with other educational networks, and mutual reference can be carried out.

2.2 The contents-characteristic of subjects

The courses designed by the DTTS were culture subjects of 11. Generally, the contents of culture subject in a training course are unlike 'learning subject' that gains new knowledge. The culture subjects are mainly implicated that the contents of knowledge or skill newly asked for with a social change. And it takes into consideration that learning environment is being home, designed so that it might participate in learning not passive position but positively.

1) Show many concrete examples so that positive and concrete study can be performed.
2) Show or introduce the newest data and the newest present condition. And a learner performs creation of a report, discussion, and practice based on this.
3) In order to check rationally learning process which is the blind spot of home study, a small-scale subjectivity formula or report is required of an evaluation item.

4) The teaching contents are selected based on an opinion of the highest specialist of the field.

2.3 The learner-characteristic of in-service teacher and consult the needs analysis

In designing we considered the needs analysis of teacher needs[2]. And also considered the spirit of teacher training reform, that is the open educational opportunity, the learning speed, the learning contents, and the learning method, etc.

3 Proposed Teaching Models

3.1 Problem-Solving type Model

This model is used the following three subjects with "understanding of tradition culture", "information society and a computer "and "environment and education". The characteristics of contents of these subjects have much problem socially now. For example, the latest children cannot have understand about traditional culture, and do not understand value either. Moreover, although environmental problems are scattered in the familiar place, the problem consciousness does not exist. It is the learning which considers how it is efficiently introduced, how solving these problems at an educational field.

Problem-Solving type model is shown in **Figure 3.1**.

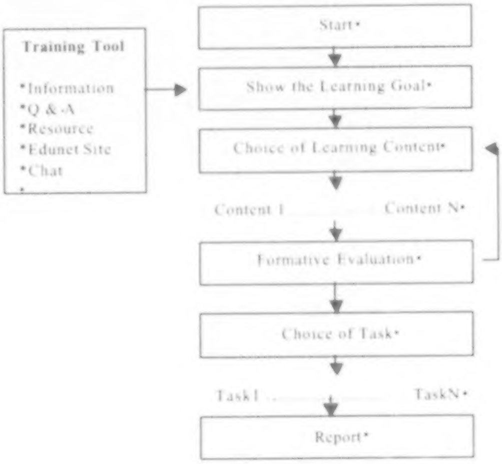


Figure 3.1 Problem-Solving type Model

3.2 Lecture-Practice type Model

Two subjects, "An information society and multimedia education" and "Theory and practice of open education" used this model. It is designed so that it might practice how theoretical knowledge may be reflected in the actual educational field. Through these courses, teacher can to help a child learn the capability that it can count measure to an information society, and how a teacher should just utilize the concept and the technology of multimedia for lesson activity. And more recently, it often pleads the open education. While introducing the concept of the open education and the example of the practice, teacher also gives an opportunity to consider an educational-practical use proposal directly.

3.3 Courseware type Model

Since three subjects, "Foreknowledge of the future society and a counter plan", "The world in the 21st century and the Korea", " Raising of national morality nature" were the contents of the type learned as new knowledge.

After having chosen the learning unit from the table of the learning contents, and learning using various data, composition which finishes a course through formation evaluation and generalization evaluation was designed.

3.4 Seminar type Model

This model uses the following three subjects. That is "The direction of the educational system reform and school reform", "Education of humanity and originality", "Education of a unification counter plan". At first a group is constructed by the theme and to be performed learning in Seminar form so that learner might have an opportunity to expand the view and develop the main point by the mode of opinion exchange.

Seminar type model is shown in **Figure 3.2** below.

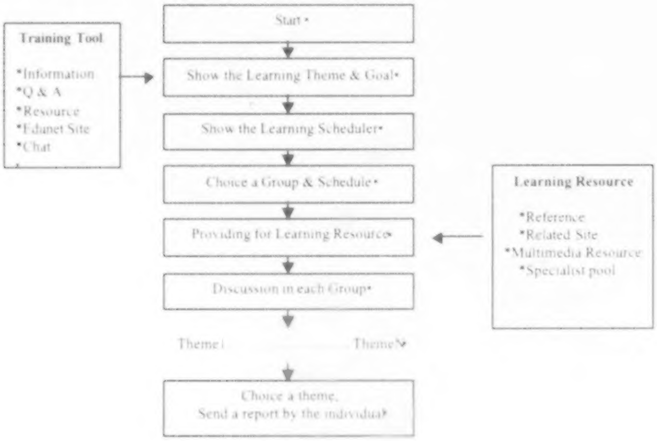

Training Tool
*Information
*Q & A
*Resource
*Edunet Site
*Chat
Start
Show the Learning Theme & Goal
Show the Learning Scheduler
Choice a Group & Schedule
Providing for Learning Resource
Discussion in each Group
Theme1
ThemeN
Choice a theme,
Send a report by the individual
Learning Resource
*Reference
*Related Site
*Multimedia Resource
*Specialist pool

Figure 3.2 Seminar type Model

4 Implementation

The proposed model went into test implementation from October 1998. And now the model is used for qualification study of elementary and the 1st class positive teacher of middle, and general training of an elementary deputy schoolmaster.

As problems of this teaching model the following may be mentioned: 1) Poorness of course contents, 2) The impossibility of checking a learning process, 3) The insufficiency of feedback to a learner etc.

5 Conclusions

The distance education which used the high technology of information communication engineering in Korea started in 1997[3]. Insufficiency of a lecturer and restriction of a training opportunity are well said as a problem in teacher trainings. As one proposal for solving this problem, the project of "Development of a distance teacher training system" started and virtual teacher training actually started from October 1998. Thereby, little by little, teacher training environment becomes better and we think that the opportunity of training and the improvement in qualitative teacher training may also be anticipated.

There are problems that should still be correct and complement continuously in this training system. But the problems that should solved urgently are preparing the method of evaluation, the monitor staff who helps training, and a specialist pool.

References

[1] Bijan B. Gillani & Anju Relan., "Incorporationg Interactivity and Multimedia into Web Based Instruction", Web-based instruction, Badrul H. Khan, (Eds.), Educational Technology Publications, Inc., pp. 231-237, (1997).

[2] Choi, Sang-gun., "Development of Distance Teacher Training System", Seoul: Korea Education Research Institute, (1995).

[3] C. H. Lee., "Pre-service & In-service Teacher Training for Computer Education in Korea," Advanced Research in Computer and Communications in Education. G. Cumming et al.(Eds.), IOS Press, pp. 433-440, (1999).

[4] Mark Grabe & Cindy Grabe., "Integrating Technology for Meaningful Learning", Houghton Mifflin Company, pp.32-76, (1998).

[5] Pang, Myung-suk., "Teacher's Information Literacy Training in the republic of Korea", Seoul; Korea Multimedia Education Center, (1998).

[6] Willis, B., "Instructional development for distance education," (ERIC Document Reproduction Service NO. ED 351 007), (1992).

Design of Multiple Metaphors in User-Interface

Yu-chen Hsu
National Tsing Hua University, Taiwan
arbia@ms55.hinet.net

As computer systems have become more sophisticated, several researchers have proposed the use of multiple models or metaphors to design computer systems and have argued that the provision of multiple metaphors would better match the characteristics of complex target systems. Multiple metaphors refer to the use of two or more distinct metaphors, each explaining various aspects of the target system. A multiple-metaphor interface means the combination of two or more metaphors to design the interface for a computer system. Although we see the strength of using multiple metaphors in interface design, not many guidelines for selecting and combining metaphors in creating a system are available. Because of the present vogue for interface metaphors and the limited research available in this area, there is no accepted standard for designing metaphorical interfaces. In this paper, the theoretical basis for the use of multiple metaphors is discussed. The method in designing metaphorical interfaces is proposed as the author created the interfaces for a metaphor study. Issues of metaphorical interface design are discussed and detailed procedures in generating and combining metaphors in creating interfaces for a hypermedia system are presented.

Keywords: **Multiple metaphors, analogy, structural cues, interface design guidelines**

1 Introduction

Metaphor is traditionally a concept belonging to the fields of linguistics and philosophy. In 1980, Lakoff and Johnson [1] presented new ways of thinking about metaphor regarding everyday experience. They consider the essence of metaphor to be "understanding and experiencing one kind of thing in terms of another" (p. 5). They demonstrate that people's conceptual systems are metaphorical and people's actions every day are a matter of metaphor. In human-computer interaction, a metaphor is "present when terminology or concepts from a familiar non-computer domain are used to depict computer functions and objects in a user interface" [2]. Two components of a metaphor are the base domain and the target domain. The base domain is "the area of knowledge or expertise which a person already possesses," while the target domain is "the area of knowledge or expertise in which a person is trying to become familiar" [2]. When a computer user learns a new computer program, he calls upon his prior knowledge (analogies or metaphors) as the basis on which to form a new mental model. Designers can take advantage of users' existing mental models to present ways of conceptualizing computer functions and to design interfaces for computing systems [3]. Carroll, Mack, and Kellogg [4] describe the use of metaphor as a way to control the complexity of user interfaces by designing the actions, procedures, and concepts of new interfaces based on users' familiar actions, procedures, and concepts from previously learned interfaces. Metaphors used in this way are called interface metaphors.

Because humans learn new concepts or knowledge in terms of what they already know, almost all the computer interfaces in the world contain various types of metaphors taken from other domains. The current concern in computer interface design for ease of use, encourages the use of familiar objects and icon-based operations. Using these, computer users do not have to interact with command line syntax like they did before. This trend also facilitates the growth of the Internet [5]. Ratzan [5] argues that "metaphors may themselves suggest further implications, inferences or interactions of computer components. Metaphors help make sense of the online environment" (p. 47). Nevertheless, design guidelines derived from research findings on Internet metaphors are far behind their present demand in the practitioner field.

2 The use of multiple metaphors in interface design

While metaphor is useful for helping people to reason about new knowledge, mismatches between the base and target domains may occur, which lead to incorrect inferences. Carroll, Mack, Kellogg [4] claim that metaphor must provide incomplete mappings to their target domains. "If a text editor truly appeared and functioned as a typewriter in every detail, it would be a typewriter" (p. 69). Mismatches happen in situations in which the attributes and relations of a single metaphor can not be perfectly correlated with the attributes and relations of the target domain. This is especially true when the target domain is so complex that no individual model can fully explain anticipated behavior. In this case, the use of multiple metaphors to design interfaces may be a solution [6]. Benking and Judge [7] explain that three or more complementary metaphors may be used together in order to adequately represent some complex systems. Booth's [8] argument provides a basis for the use of multiple metaphors. He notes that people "appear to have blocks of knowledge relating to different domains and use parts of these knowledge blocks when they believe that it is appropriate" (p. 73). One kind of composite metaphor (multiple metaphor) is the use of complementary metaphors to represent functions of an interface, with each metaphor representing a function at a single level. Carroll et al. give an example of this kind of composite metaphor -- the integrated office system, which includes electronic mail, spreadsheets, text editing, and decision support, each with a different metaphor to represent it in the system.

In terms of the theoretical basis for multiple metaphors, Rumelhart and Norman [9] conducted a study on teaching new users to learn a text editor. They observed that students made errors because of their inadequate conceptualization of the various parts of the computer system. This resulted from the insufficient mental models students brought to the situation; they limited the kinds of analogies they might have employed. The authors note that no single metaphor can fully explain a complex piece of subject matter. Thus, they postulate an effective solution to eliminate student errors -- the provision of a more appropriate analogical framework, with different conceptual models to help students in their reasoning. They developed the "secretary," the "card file," and the "tape recorder" models, each explaining various aspects of the text editor, and claim that, although none of the models are perfect, as people grow more experienced in a domain, they become better at choosing appropriate models for a specific situation. In teaching this subject matter, it is effective to present a set of models, each with their own built-in context dependencies, as alternative conceptualizations of the target domain.

Collins and Gentner [10, 11] found that analogies allow people to create multiple mental models for use in reasoning about a complex system. They discuss Gentner's [12] analogy hypothesis and note that "a major way in which people reason about unfamiliar domains is through analogical mappings" (p. 247). They used analogies to map the set of transition rules from a known domain (the base) into the new domain (the target), thereby constructing a mental model that can generate inferences in the target domain. To test this hypothesis, they observed how subjects reason about evaporation and did an analysis of their protocols. The qualitative data suggests that subjects formed three different levels of interrelated mental models in reasoning about the target domain. These findings support the view that people learn the target domain by partitioning it into different component models, each mapped to a different base domain.

Multiple metaphors have also been employed in the field of artificial intelligence. Burstein [13] presented a model for students to learn a programming language. This involved the use of a box analogy, an algebra analogy, and a human processor analogy. In the example, the author used analogies in such a way that each analogy covered several levels of description, but served different functions. The use of multiple analogy models has been found to be more helpful in facilitating students who are learning the new domain in this case.

Spiro et al. [14] describe the danger of using single analogies in learning and instruction. They suggest that misconceptions are often caused by the reductive effect of analogies. "When analogies are used to 'start simple,' the knowledge ultimately acquired often stays simple. Well-intended analogies often result in oversimplified knowledge" (p. 502). They present eight situations in which the use of an analogy induces misconceptions or mismatches. One common characteristic of these eight situations is that users tend to depend too much on the properties of an analogous source domain in understanding the topic (target) domain. To solve this problem, they propose an antidote -- the use of integrated multiple analogies to represent complex concepts. They claim that by introducing new analogies which emend the missing or misleading aspects of the earlier analogy, the strength of the original analogy is retained, but its weakness is discarded. To give an example, muscle fiber function is proposed as the target concept, which is then explained by three analogies -- the rowing crew, the turnbuckle, and the Chinese finger-cuffs analogies. To integrate multiple analogies, they propose the technique of "composite imaging with selective contingent instantiation" (p. 522), in which three analogy models are created separately for the comprehension of the muscle fiber function with the applicability of the elements in each analogy being context-dependent. Although the authors claim that this technique could be applied mentally or to computer graphic displays, its implications for the design of a composite metaphor are limited.

The advantage of using multiple metaphors in designing computer interfaces can be seen. However, there is no accepted standard for designing an interface with multiple metaphors. In the following section, the issues or problems of designing a multiple-metaphor interface will be discussed.

3 Issues to consider in the design of multiple (composite) metaphor interfaces

Judging from previous studies [15, 2] and my experience, I conclude a number of difficulties that designers or researchers would face in creating a multiple metaphor interface. Since the generation of a multiple metaphor interface involves the selection and combination of multiple metaphors, design considerations and problems will be discussed below within these two phases.

3.1 Selection of metaphor

When selecting metaphors to design interfaces for computer systems, the designer needs to consider several issues, which include the type of information, description level of metaphors, users' expert levels and prior knowledge, users' tasks, methods of task completion, and appearance of the interface.

- The *type and structure of information* of the target system influences how designers select metaphors. These attributes include the information content and structure of the target system. Designers need to consider the type of content information when choosing appropriate metaphors.

- In terms of *description level* of the target system, Booth [8] claims that "the level of description of a metaphor is concerned with the type of information that a metaphor might be expected to communicate" (p. 77). He takes an example from Moran's [16] Command Language Grammar and says that a metaphor can be aimed at the task level, the semantic level, the syntactic level, or the physical level. This characteristic increases possibilities, but also the difficulty in designing a metaphorical interface.

- When choosing metaphors, the designer should consider *users' prior knowledge* in their familiar domains as a basis for designing tools for learning new things. Stagger and Norcio [17] claims that, when designing multiple models for users to learn new knowledge, designers need to consider the expert level of the users and the tasks to be completed. As users gain expertise in the target area, their ability to manipulate multiple models increases. Since metaphors work by mapping previously acquired knowledge of users to the target domain that they are going to learn, some attributes (objects, relations, actions, effects) of the base domain must match with the attributes of the target domain. The selection of metaphors should be based on a user's familiar knowledge.

- Carroll et al. [4] explain three aspects to consider in designing a metaphor: the *tasks, methods, and appearance levels*. The task level describes users' goals and what they can do; an example is the information search in the present study. The method level describes how tasks are accomplished. The appearance level is the "look and feel of the task situation vis-à-vis the physical implementation of the domain" (p. 78). It includes aspects of the hardware and the presentation of screen objects.

In addition to consideration of the above criteria in selecting interface metaphors, designers also face some design problems described by Cooper [18]: 1) there are not enough metaphors; 2) the metaphors do not scale well; and 3) the ability of users to recognize them is questionable. As the number of metaphors increases in designing an interface, there are more constraints regarding the criteria of metaphor selection. Carroll and Thomas [3] suggest that when using two or more metaphors to design a system, one should not choose objects or procedures that are exclusively alternative to each other, so as to avoid interference and confusion. In another article [19], Carroll and Mack argue that good metaphors should also not provide completely transparent and comprehensive mapping, so that they may better enable users to learn.

3.2 Combination of metaphors

Once multiple metaphors are selected, designers need to identify an optimal way of combining the metaphors. This issue has not been well discussed by scholars, so there are not many guidelines regarding how to combine multiple metaphors to create a computer interface.

Beyond the issues discussed above, designers face some additional problems in combining multiple metaphors. First, it is hard to draw the boundaries between different metaphors. Booth [8] raises this question for designers: "how [can we] signpost the boundaries of metaphors within a system so that users know when a metaphor is no longer relevant and when another metaphor is appropriate [?]"(p. 78) Second, although the idea of using multiple metaphors has been suggested for interface design by practitioners, the way to operationalize multiple metaphors to create interfaces is very difficult to carry out. Most previous studies used separate analogies to teach new knowledge, or they used separate metaphorical interfaces to help users to learn new computer systems. Methods for combining different metaphors in a system have not been explored. Smilowitz [2] was a pioneering researcher who tried to mix two metaphors in an interface in her experimental studies. Due to the challenging nature of this case, there were design deficiencies in her approach to combining multiple metaphors. Smilowitz tried to mix two metaphors within the navigation area in a hypertext system. But navigation tools are only a part of an interface. The design in her study limits users' perceptions of the metaphorical interface.

In light of the above difficulties, the application of structural cues taken from multiple metaphors may be a solution to integrate two or more metaphors in designing an interface. The next section is a brief review on structural cues in computer interfaces.

4 Hypertext structure cues

Dillon [20] presents a discussion on the structure in documents. He argues that the meaning of structure differs depending on different standpoints: from the perspectives of writers, readers, or from the consideration of reading/writing tasks. There is a difference in the structures of a paper and an electronic document. Compared to a paper document, a hypertext document does not have the same amount of information available to the readers, and its structures do not have equal transparency. In a hypertext system, the author can create numerous structures from the same information. Due to the lesser experience novice users may have with hypertext systems, "users' schemata of hypertext environments are likely to be 'informationally leaner' than those for paper documents" (p. 114). These reasons may explain why users easily become lost in hyperspace.

One way to solve navigation problems in hypermedia is to provide structure aids that inform users of what information is available, as well as where it may be located and how it may be organized. Hulley [21] discuss hypermedia and notes that "its structure needs to be made obvious to the users [,] and a means of browsing and navigating around it needs to be provided." In addition, he thinks that "the methods chosen for structuring information need to be the most suitable for the user's needs; they must support the tasks that the user wants to carry out and provide an interface which can be easily understood or learned" (p. 173). Thuring, Hannemann, and Haake [22] argue that the coherence of a hyperdocument has an impact on the reader's information processing. Well designed hypertext structures plus the presence of rhetorical cues may facilitate coherence; so designers should provide cues at both the node (within nodes) and net (between nodes) levels. Rouet and Levonen [23] describe the prototypical representation of hypertext as "a set of text units connected through multiple links, that is, a text network" (p. 15). Due to the navigation problems which a novice user may experience, they argue that novice users need analogies with conventional structures.

5 Procedures used to create multiple-metaphorical interfaces

From the literature review we know that a single metaphor does not cover everything in the interface. The use of two or more metaphors in designing an interface may be a solution for the problem of mismatch and may better represent the elements as well as the relations to the target system. In addition, structural cues or metaphors should be provided to hypermedia users for them to understand the way to navigate the system.

Because of the present vogue for interface metaphors and the limited research available in this area, there is no accepted standard for designing metaphorical interfaces. Interface design in this area is more laborious because of this problem concerning the operational definition of metaphorical interfaces. Due to the problems designers or researchers may experience when combining multiple metaphors to create an interface, the creation of a metaphorical interface by combining structural cues that are derived from two or more metaphors may be a useful way to help users to search within a hypermedia system.

In my metaphor study, I compared single versus multiple-metaphor interfaces on their effects in facilitating users' information search behaviors in a hypermedia system. There were three interfaces as the independent variable in the study, with the first interface containing less metaphorical elements, the second containing some metaphorical elements from a single metaphor, and the third (multiple-metaphor interface) containing more metaphorical elements from two metaphors.

A method was proposed to create the metaphorical interfaces based on existing design guidelines and the revision of design methods used by other researchers. In this strategy, the metaphors work as the source for the structural cues to be combined in creating a metaphorical interface. Metaphors were used as the basis for deriving navigational cues, but those cues were not treated illustratively. In other words, the cues are related structurally to the metaphors but do not necessarily represent elements of the metaphors in a pictorial way. The metaphors used provided a logic for the designs, which guided the choice of structural cues that distinguish the three interfaces by the varying degrees to which they appear. This approach leads to a more precise operational definition and manipulation of the variables.

A detailed review of the method for creating the three interfaces for the experimental study is presented in the following paragraphs.

5.1 Selection of context and task for the study

In the study information searches were chosen to be the user task in order to investigate the effectiveness of using metaphors as a navigational aid in designing a hypermedia system.

5.2 Selection of information content (information system)

After the task had been chosen, the information content was identified based on the scope and structure of the system, the depth and width of the information structure, characteristics of the information related to the target system, the familiarity of the content to the potential subjects, the availability of the system, its ease of implementation, appropriate metaphors for this information content, and software used to create the hypertext system. These issues were identified partially through a review of articles describing the processes that other designers went through to create metaphorical interfaces and the criteria they took into account [24, 25, 26, 27, 28, 29]. The web sites of some universities, the web pages of the Library of Congress and the National Science Foundation, a CD-ROM in science, a health database, geography information, the lives of musicians, world art and wars, and American history were reviewed. Some systems were good because they have structures embedded within them for information searching and navigation; but there were no appropriate metaphors for those types of content. For the other systems, there were appropriate metaphors that matched the content or elements, but the information structures were not useful for information searching.

The CD-ROM "The Enduring Vision" [30] was finally chosen as the content on which to base the hypertext system. It contains 33 chapters of American history. The same content as that in the CD-ROM can be found in the book The Enduring Vision, and the CD-ROM structure is similar to the structure of the book. This similarity between the book and CD-ROM was a positive factor which influenced my choice of the CD-ROM. It made the job of creating the three variances of hypertext easier, because the content did not have to be restructured to fit the book metaphor. Four chapters with a total of ninety-eight articles in nineteenth-century American history were selected from the program to create the hypertext system.

5.3 Selection of metaphors

The chosen hypertext system placed constraints on the variety of possible metaphors that could be used to design the interfaces. The issues taken into account at this stage included the subjects' prior knowledge concerning the metaphors, the characteristics of hypermedia systems versus the attributes of metaphors, the overall structures of metaphors in covering hypermedia systems, the appropriateness of metaphors for information searching, potential mismatches between the metaphors and the hypermedia system, existing metaphors used in other software, ease of representation, manifestations /appearances of metaphors, guidelines for metaphor design, and methods of combining multiple metaphors. In addition, the characteristics proposed by Lin [15] were also taken into consideration. These are: (1) the style of presentation of information, (2) the size of information units, (3) the degree of user control over the ordering of information, (4) routes of traversing, (5) the visibility of linkages among units, (6) the implied internal structure of information units, and (7) the style of access to specific information. Some other design guidelines were also taken into consideration when I created the metaphorical interfaces. After several unsuccessful attempts (using different combinations of multiple metaphors such as timeline, map, journey, path, container, building...), the book and folder metaphors were ultimately selected, based on the above guidelines and the consideration of possible ways of metaphor combination.

One criterion in selecting multiple metaphors is that the chosen metaphors must be independent of each other. In other words, one metaphor can not be a secondary (subordinate) metaphor to the other one (the primary metaphor). According to Cates [24], a primary metaphor refers to the principle or first metaphor employed, and a secondary metaphor means a subsequent metaphor employed. The secondary metaphor "stimulates images and semantic expressions related to those stimulated by the primary metaphors which they are intended to accompany."(p. 98) If one metaphor is subordinate to the other, then they can be seen as the same metaphor.

The reason I selected book and folder metaphors is that each could map to different aspects of the hypermedia system, so they are complementary (see Table 1 for analysis of metaphor functions). Spiro et al. [14] propose the employment of multiple analogies in learning and instruction. They identify eight ways that analogies may induce misconceptions. Based on their framework, I analyzed the strengths and weaknesses of book and folder metaphors and used them in the design of three interfaces. Due to the scope of this paper, this analysis will not be discussed in the current paper.

Table 1: Analysis of metaphor functions and structural cues

	Elements related to content or to the hypermedia structure	**Metaphor functions related to hypermedia characteristics**
Book metaphor	• Books have structure. Each book page can be used to place an information unit (e.g. a web page) • A book has a cover, a table of contents, chapters, sections, and pages • Turn page/ page number • Open/close a book	• There is visibility of linkages among units • A book structure implies the internal structure of information unit in a hypertext system • A book has a particular style of information presentation and access to specific information • Book pages need to be accessed sequentially by page turning or by reading the table of contents (Lin, 1989, p. 48)
Folder metaphor	• Each folder can be used to place an information unit • Folder tabs with labels enable users to easily identify the content of a folder • Flexible ordering of information (Lin, 1989, p.46)	• Folder tabs allow random access to specific information units (Lin, 1989, p. 46) • Information units in different levels can be directly accessed

5.4 Combination of two metaphors and the creation of three interfaces (use of structural cues)

After each metaphor was selected, all of their objects and functions were analyzed using the POPIT model as shown in Table 2 [24]. The design problems which previous researchers faced in combining multiple metaphors in one interface (the creation of a composite metaphor interface) have been discussed in previous sections. In addition, Lakoff and Johnson [1] claim that "metaphors do not imply a complete mapping of every concrete detail of one object or situation onto another; rather they emphasize certain features and suppress others" (p. 96). It is also impossible to manipulate metaphorical elements in an interface from complete absence to presence.

In view of these difficulties, the three metaphorical interfaces were created in such a way that each interface contains various degrees of structural cues taken from one or two metaphors, with the cues ranging from minimum to maximum. Rather than call them the no-metaphor, single-metaphor, and multiple-metaphor interfaces as in previous studies, they were called "the interfaces with different degrees of structural cues derived from single or multiple metaphors."

Table 2: POPIT Model (Cates, 1994)

	Book (Cates, 1994)	Folder
Properties	Book cover, pages, table of contents, chapters, sections, title	Tabs on the top of each folder, several folders can be put together, labels or keywords on the tabs
Operations	Open or close a book, turn page forward and backward	Folder tabs can be thumbed through, it allows random access, flexible ordering of folders
Phases	Turn pages, open, close, begin reading, find information/words, highlight	Select the folder, open the folder, select section by way of thumb tabs
Images	Book cover, layout of books	Same size and shape for each folder, tab in different location on the top, tabs have different colors
Types	Reference works, recreational books	Information storage

First interface (interface A with minimal cues)

There were a total of one hundred screens in this interface. Interface A contains a minimal degree of metaphorical concepts. In addition, the three interfaces in this experimental study needed to have similar structures and styles of information presentation for the sake of comparison and data collection. Since all computer interfaces contain some degree of metaphors taken from other domains, it is impossible to rule out metaphor elements totally. In the hypertext system, each article was organized hierarchically with its title listed in previous levels as hyperlinks. This made the title lists look more like the content lists of a book. For this reason, interface A still contains a small number of metaphorical elements, partially due to the nature of the information content.

Second interface (interface B with medium cues)

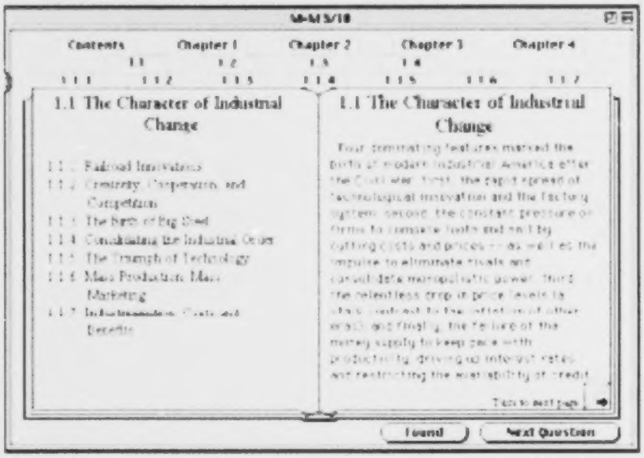
There were a total of four hundred and twenty seven screens in each of the interfaces, B and C. The same structural components and elements can be found in all three interfaces: 1) four levels in the system, 2) a main page as the first level with hyperlinks linking to the second and the third level, 3) articles in the second, the third, and the fourth levels with or without hyperlinks linking to the next level, and 4) titles and body text for each article. Based on the analysis, terms, images, structures, and operations were taken from a book metaphor to add to the design of the second interface. This causes the second interface to contain more structural cues from a book metaphor. All the information was presented in book format; for example, the text of each article was presented on double-sided pages, as in a real book. Users can click on the dog-ear to turn to the previous or next page. In addition, the title of each article was labeled with chapter, section, and subsection number to resemble the title of a book.

Third interface (interface C with maximum cues)

In order to compare the effects of the interface in which the structural cues were taken from only one metaphor with the interface in which structural cues were taken from multiple metaphors, a third interface was created (see Figure 1). Extra structural cues, including images, structures, and operations of a folder metaphor, were added to the second interface to create the third one. Whereas information presentation in a book metaphor is linear, the folder metaphor conveys the hypermedia attribute of flexible information access. The book metaphor worked as the main metaphor and was broader in its scope, while the folder metaphor was added to the design to supplement the book metaphor. Booth [8] describes the dimensions of a metaphor in terms of its scope and level of description. The scope describes the number of concepts that a metaphor addresses, and the level of description deals with the information types that a metaphor communicates. Similarly, Hammond and Allinson [30] describe four levels of information that a metaphor may convey: task information, semantic information, lexical information, and physical information. Using those concepts to examine the design in the present study, the book metaphor has a larger scope and it conveys four information levels: the hypertext structure, the layout, the terms, and the operations; but the scope and information levels of the folder metaphor are more restricted. The main function of the folder metaphor is to provide a flexible means of information access so that users can randomly access articles in different levels.

The structural cues taken from the book and folder metaphors include content lists, section titles and number, double-paged layout, book turning corners, folder labels with section numbers, physical layouts of book and folder metaphors, and so on. Those elements consist of textual and graphical structural cues, which were combined to create the metaphorical interfaces.

Figure 1. Screen-shot of the third level for interface C (multiple-metaphor interface)


Contents
Chapter 1
Chapter 2
Chapter 3
Chapter 4
1.1 The Character of Industrial Change
1.1 The Character of Industrial Change
Found
Next Question

6 Conclusion

Metaphors do not apply equally in the interface designs, and usually only the most salient points are drawn from a metaphor. The book and folder metaphors are not alternative choices; instead, they complement each other (in a non-exhaustive way). This is consistent with Benking and Judge's [7] view of using three or more complementary metaphors to explain complex systems. Due to the consideration of ease of manipulation in the experimental study, only two metaphors were chosen for the design of the interfaces. At each step in creating the interfaces, not only more possibilities, but also a few constraints were added to the design. The selection and combination of the structural cues did not result in perfect designs, but they were completed with much deliberation concerning the many design possibilities and tradeoffs.

Designing metaphorical interfaces involves many other issues that are beyond the scope of the present discussion. The compatibility between the metaphors to be combined may play an important role in appropriately conveying the functions of each of the interface elements. Interfaces that are created with incompatible metaphors could cause misunderstanding and hinder users' performances. The problems of selecting and combining multiple metaphors have been discussed in this paper, and the procedures for creating the metaphorical interfaces have been explicitly presented. There are many complex design issues involved in the creation of metaphorical interfaces. The design of a metaphorical interface relies on appropriate metaphor selection and combination in order to achieve optimal effects. Further research is needed to explore other possible ways to combine multiple metaphors to create user-friendly interfaces.

Due to the scope of this paper, the whole process of creating metaphorical interfaces can not be discussed in detail. However, it is the hope of the author that the method presented can provide interface designers and researchers with insight in creating metaphorical interfaces.

References

[1] Lakoff, G., & Johnson, M., Metaphors We Live By. Chicago, IL: University of Chicago Press, (1980).
[2] Smilowitz, E. D., Metaphors in user interface design: An empirical investigation. Unpublished doctoral dissertation, New Mexico State University, (1995)

[3] Carroll, J. M., Thomas, J. C., Metaphor and the cognitive representation of computing systems. IEEE Transactions on Systems, Man, and Cybernetics, SMC Vol. 12, No. 2, pp. 107-116, (1982).

[4] Carroll, J. M., Mack, R. L., & Kellogg, W. A., Interface metaphors and user interface design, In Helander, M. (Ed.), Handbook of Human-Computer Interaction (pp. 67-85). North-Holland: Elsevier Science Publishers, (1988).

[5] Ratzan, L. D., Maing sense of the Internet: A metaphorical approach. Unpublished doctoral dissertation, The State University of New Jersey, New Brunswick, (1998).

[6] Williams, M. D., Hollan, J. D. & Stevens, A. L., Human reasoning about a simple physical system, In D. Gentner & A. L. Stevens, Eds, Mental Models, pp. 134-154, Hillsdale, NJ: Lawrence Erlbaum Associates, (1983).

[7] Benking, H., & Judge, A. (1994). Design Considerations for Spatial Metaphors [On line]. Available HTTP:

[8] Booth, P. A., An Introduction to Human Computer Interaction. Alwrence Erlbaum Associates, Publishers, Hillsdale: C4, p65-101, (1989).

[9] Rumelhart, D. E., & Norman, D. A., Analogical processes in learning. In J. R. Anderson (Ed.), Cognitive Skills and Their Acquisition (pp. 335-361). Hillsdale, NJ: Lawrence Erlbaum Associates, (1981).

[10] Collins, A., & Gentner, D., Multiple models of evaporation processes. Proceedings of the Fifth Annual Conference of the Cognitive Science Society (pp. 16-20). San Diego: CA, (1983).

[11] Collins, A., & Gentner, D., How people construct mental models. In D. Holland, & N. Quinn (Eds.), Cultural models in language and thought (pp. 243-268). New York: Cambridge University Press, (1987).

[12] Gentner, D., Are scientific analogies metaphors? In D. S. Miall (Ed.), Metaphor: Problems and Perspectives (pp. 106-132). Atlantic Highlands, NJ: Humanities Press, (1982).

[13] Burstein, M. H., Concept formation by incremental analogical reasoning and debugging. Proceedings of the International Machine Learning Workshop (pp. 19-25). Monticello, IL: University of Illinois, (1983).

Designing A Web-Based Action Learning Environment - Integrating Learning and Working in One Environment

Feng-Kwei Wang* and John Wedman**
**University of Missouri - Columbia*
221-I Townsend Hall, Columbia, MO 65211, USA
Tel: +2-573-8843918
Fax: +2-573-8844944
wangfeng@missouri.edu
***University of Missouri - Columbia*
303-H Townsend Hall, Columbia, MO 65211, USA
Tel: +2-573-8824546
Fax: +2-573-8844944
wedmanj@missouri.edu

Action learning has become a popular approach to management education. Many advocates of "work while you learn and learn while you work" define action learning as an experience-based approach to developing workers by integrating real workplace problems and dilemmas in development programs as a way to work and learn concurrently. In tandem with the increasing acceptance of action learning, Web-based learning has become a common practice in both school and work settings. The authors contemplate that Web technology be utilized to build a Web-based action learning environment. In this environment, the tools and resources selected and devised are intended to facilitate the pedagogical processes of action learning. The biggest advantage of this environment is that learners are able to conduct action learning without constraining to time and space boundaries. The authors also argue that any technology-based learning environment would be flawed without a sound design framework and strong cultural and leadership supports for implementation.

Keywords: **Action Learning, Web-based Learning Environment, Knowledge Repository**

1 Introduction

Recently, learning based on socio-cultural theories such as situated learning [1] and cognitive apprenticeship [2] have gained attention in academic education and corporate training. There are three core doctrines of socio-cultural learning theories [2,3,4,5]. First, knowledge is situated in meaningful tasks that the learner carries out. Second, learning is a social process in which the learner interacts with peers or experts. Third, mediated tools and signs in the socio-cultural milieu of the learner affect learning processes and results. In other words, these theories hold that learning occurs in a socio-cultural context where the learner carries out authentic tasks. Thus, it becomes essential to combine working and learning contexts in the workplace if learning is to be effective. In parallel with the development of socio-cultural learning theories, action learning has become an emerging paradigm for workplace learning [6,7]. The tenet of action learning is learning through action [8,9]. In action learning, learners learn with and from each other, with the help of facilitators, by working on real problems and reflecting on their own experiences.

In addition to the emerging learning theory that has changed today's landscape of learning design and implementation, learning technology is another area that has a deep impact on the construction of learning environments. The technological advancements that the Internet and the Web have brought about have outpaced pedagogical and human learning theories [10]. The interactive, distributed, and collaborative features the Web offers present unprecedented opportunities for experimentation in creating an online learning environment that allows learning to occur without temporal and spatial constraints. Learning anytime, anywhere through the Web has become a common practice in the educational circle [11].

This paper intends to provide the design framework of a Web-based action learning environment (WALE) that combines contemporary learning theory and learning technology to facilitate and support the process of action learning. We begin with an overview of action learning and groupware technology, which a WALE is built upon. Then, we introduce the design framework of a WALE. We also present an actual WALE we built to give readers a better understanding of this WALE framework.

2 Action Learning

The concept of action learning made its appearance in 1954 as a method of problem solving. In 1967, Revans first implemented action learning in Belgium as a method of management development. Revans [12] defines:

> "Action learning is a means of development, intellectual, emotional or physical, that requires its subject, through responsible involvement in some real, complex and stressful problem, to achieve intended change sufficient to improve his observable behaviour henceforth in the problem field" (p. 4).

According to Revans [9], learning (L) is the sum of acquired programmed knowledge (P) and questioning insight (Q) and is denoted as:

$$L = P + Q$$

P represents the traditional instruction received in formal academic institutions such as business schools, and is deployed by experts. Q represents one's own findings from managerial experiences and is exercised by leaders. It should be noted that P and Q are complementary parts of a total development process [13, 14].

In a typical action learning program, programmed instruction might be given on a designated theory or theoretical topic. In conjunction with the programmed instruction, learners are asked to apply their prior and new knowledge to a real project that organizational sponsors have sanctioned. Throughout the program, learners continue to work on the project with assistance from qualified facilitators or advisors as well as other learners who help them make sense of their project experiences.

Action learning situates learners in a real-life problem in which they learn from and with others as they solve organizational problems. While there are many variations of action learning, Beaty, Bourner & Frost [15] argue that four essential elements of action learning are

- real problems,
- group reflection,
- personal responsibility, and
- action based

First, learners learn from tackling organizational problems they face within their work contexts. Second, learning is a social interaction in which learners learn with and from a group of others who are also engaged in managing real problems. Third, members of the group are accountable for solving their own organizational problems. Fourth, action learning does not stop with theoretical solutions. It is concerned with implementing the actions that the group has explored.

Action learning holds that learning for learners means learning to take effective actions [14], which only occurs when learners actually engage in taking actions. The best actions for learning are those that solve an organization's real problems, those actions that are significant to the learners themselves. The learning process is a social interaction in which a group of learners work together as a team on the problems. The learners learn best with and from one another through peer interactions and discussions.

"Action learning is holistic in its view of the person [learner], the management process, and learning. It is highly situational, flexibly treating elusive problems and combines a social process with individual needs" [14, p37]. Its value lies in the situated characteristics of knowledge and skills acquisition. Through hands-on experiences with peers in solving real-life problems, learners can develop their own theories of learning and management in action, which are tested against real-world experiences as well as established tenets [16]. Learners are able to sharpen their problem-solving, communication and critical-thinking skills and to build skills that are germane to their own particular organizational needs. Furthermore, action learning, in a broader sense, has much in common with the concept of the learning organization [14]. The critical features of action learning are in accordance with the five disciplines of the learning organization: system thinking, personal mastery, mental models, building a shared vision, and team learning [17]. From this perspective, action learning is not only a matter of individual learning and action but is also an organizational transformation process that deals greatly with organizational dynamics and culture.

3 Augmenting Action Learning with Technology

While action learning is taking off and is proven to be effective in management education [18, 19], we believe that Web technology can augment its effectiveness. First, the use of Web technology to communicate organizational information, to coordinate workflows, and to collaborate on work tasks becomes indispensable in solving today's organizational problems. Since action learning emphasizes that learning comes out of business actions, we believe Web technology is instrumental for learning, in that the common Web functions can concurrently support working and learning. Second, the most vital resource in action learning is the participants' own experiences and resources. The sharing of these experiences and resources often occurs only when participants convene in action learning meetings where the majority of problem diagnosis, group discussion, solution planning, and collective reflection take place. Consequently, these valuable experiences and resources are not captured, widely disseminated, or even lost outside action learning meetings or programs. Third, time and geographical boundaries often put constraints on where and when action learning meetings can occur. The communication functions of Web technology such as e-mail and online bulletin boards provide the means to break the limitations of space and time. These functions enable the continuity of learning process beyond face-to-face meetings. Furthermore, the collaborative features of Web technology can be used to engage people in the action learning process. Although well-designed action learning programs do a good job of involving participants in learning and action, the ability to let people collaborate anytime, anywhere creates an expectation that action learning is a collective effort and every participant is contributing.

4 Web-based Action Learning Environment

With the characteristics of action learning and Web technology in mind, we develop a framework for designing a Web-based action learning environment (WALE) as shown in Figure 1. In this framework, learning occurs when learners engage in action learning to solve organizational problems with the support of Web tools and resources. The tools and resources selected and devised for the learning environment are intended to facilitate the pedagogical processes of action learning. The design of action learning and technological support has to take into account the organizational context and should be constantly evaluated and improved accordingly. The Web tools and resources are devised into three categories: knowledge repository, collaborative tools, and cognitive tools.

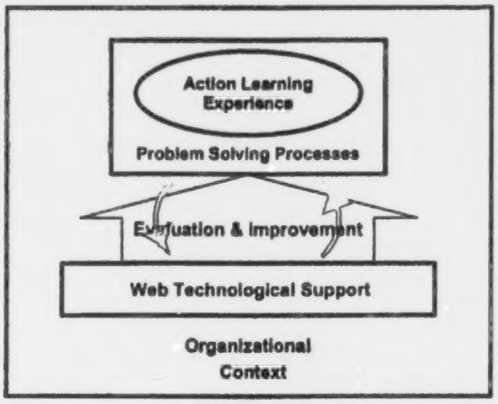

Action Learning Experience
Problem Solving Processes
Evaluation & Improvement
Web Technological Support
Organizational Context

Figure 1. The design framework of a Web-based action learning environment

4.1 Knowledge Repository

At the conceptual level, a knowledge repository is about capturing and preserving the theory and practice of practitioners in an organization. The theory component represents what Ravens called programmed knowledge (P). In action learning, there is much in theory that can inform action. For one thing, it allows practitioners to see problems in a new light. Further, it might even reveal problems undiscovered for lack of recognizable solutions. The practice component is Revans' Q that represents practitioners' own findings from their experiences. These experiences are transformational and knowledge-based in a way that is useful to an organization. They provide the means of organizational learning, from which organizational members can gain insight and understanding. In action learning, experiences edify the program participants' past success and failure of actions. They also provide the questioning insights upon which the participants can reflect and guide their future actions.

At the detailed level, a knowledge repository is a collection of electronic documents that contains basic concepts in a subject domain and extracted experiences from practitioners including cases, lessons learned, best practices, techniques, tips, references, and other knowledge granules with powerful searching functions and easy navigational tools.

4.2 Collaborative tools

Collaborative tools, which include computer conferencing, electronic mail, and shared workspace, are used to promote collaboration among participants in an action learning program. Participants take on problem solving collaboratively through this online environment built on computer networks. Through the networks, multiple perspectives and diverse learning approaches can be stimulated, with each reinforcing the others [20]. Computer conferencing permits the development of online, asynchronous, many-to-many person discussions. Electronic mail allows each participant to send messages relating to personal issues to a specific person or group [21]. Computer conferencing and electronic mail extend the time and space boundaries of action learning beyond action learning meetings and moves learning directly into the workplace. They enable action learning anytime, anywhere and make action learning an ongoing process. The shared workspace serves as the group memory, recording group activities and information in action learning. It is capable of tracking a participant's or a group's action patterns and learning paths, which indicates what actions have been performed and what information has been accessed. Each participant can either reflect on his or her own action learning history or can learn from others by reviewing the group processes.

4.3 Cognitive Tools

Kozma [22] explains that the computer can alleviate the learner's information processing burden, thereby extending human cognition. In case problem solving activities, computer tools are used to ease and enhance the performance of cognitive tasks. Such tools in a WALE include performance support, hypermedia, and navigation functions. First, performance support functions are a set of Web tools or electronic job aids that participants use to facilitate problem solving. These functions ease the cognitive load of many arduous but necessary work and/or learning tasks and make learning and problem solving more efficient. The use of problem diagnosis forms and online action learning guidelines are two examples. Second, in a hypermedia environment, knowledge is purportedly organized by mirroring the structure of human thinking. The process of imitating human thinking proceeds through associating one piece of information to a related piece of information. It functions as "knowledge on demand" and exhibits the capacity to branch from one thought to related knowledge or experiences [23]. That being so, an appropriately structured hypermedia system should be able to mirror the semantic network of an experienced or knowledgeable performer or expert [24]. Third, navigation functions such as searching, navigation maps, indices, history, and bookmarks prevent learners from getting lost in the spacious knowledge ocean and point participants in the right direction. Navigating with such tools quickly brings participants the part of knowledge that they are looking for. Navigation maps show where participants are and where they have been in knowledge repository. Similarly, indexes offer participants different ways of identifying and viewing knowledge. In contrast, the history function keeps track of navigational paths and allows participants to trace their learning processes. Finally, bookmarks register particular knowledge locations for later quick access.

5 Learning and Working in a WALE

From the process standpoint, building a WALE encompasses a set of interrelated processes that engage participants in the problem-solving activities. These processes become a way of identifying and understanding interrelated factors while helping fill gaps, minimize redundancies, and eliminate conflicts toward common goals. They enable participants to develop themselves by building, reflecting on, renewing, and sharing what they know and how they do things in solving organizational problems. In this way, a WALE integrates learning and working in one environment through: 1) online action learning activities with peers and facilitators, 2) the utilization of Web-based learning and performance support tools, and 3) full-time access to problem solving resources and results (see Figure 2).

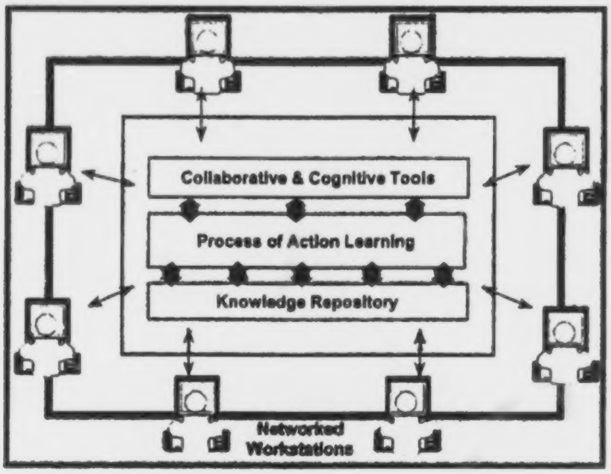


Figure 2. Working and Learning simultaneously in a WALE

6 A Case in Point

Teacher education programs have been challenged to respond to advances in technology. Unfortunately, these programs are criticized for not adequately preparing teachers to use technology in their teaching. The Office of Technology Assessment (1995) reported on technology in teacher education and noted significant limitations, including 1) faculty not modeling technology use; 2) students learning about technology – not with it; 3) field experiences not designed to model the use of technology; and 4) technology isolated from the main curriculum and pedagogy of teacher education. These limitations point to the need to revamp teacher education programs at many universities.

While teacher education faculty are central to the problem and its eventual solution, individual faculty are typically powerless to address these limitations. Deans, Directors of Teacher Education, Department Chairs, and other college and school-level leaders are best positioned to make a response, but seldom are these individuals prepared to deal with the many complexities, technical and otherwise, creating barriers to integrate technology in teacher education. Moreover, the rate of technological changes makes technology integration in teacher education a perpetual endeavor. Learning to solve problems means taking action in solving problems.

Funded by a grant from the U.S. Department of Education, a consortium of teacher education programs at the Universities of Missouri, Nebraska, Oklahoma, and Kansas, and Texas A&M was established to tackle many similar problems found in integrating technology in teacher education. This consortium is grounded with a common vision: teachers and students enabled by new and emerging technology and building a better future for all. The common mission is to better educate future generations of teachers to use technology. Recognizing that accomplishing the mission is an ongoing endeavor and the means to the end is constantly in flux due to fast emerging technology innovations and student needs, the leaders of consortium programs take the action learning approach to prepare themselves and their programs.

Adopting the WALE framework, the consortium, led by the University of Missouri, initiated a WALE development project to deploy a knowledge repository to support action learning programs engaged by respective consortium members. The Technology Integration Process (TIP) knowledge repository captures, organizes, and disseminates the collective knowledge about technology integration in teacher education, thereby leveraging the professional knowledge across many programs. Figure 3 shows the entry screen of the TIP system.

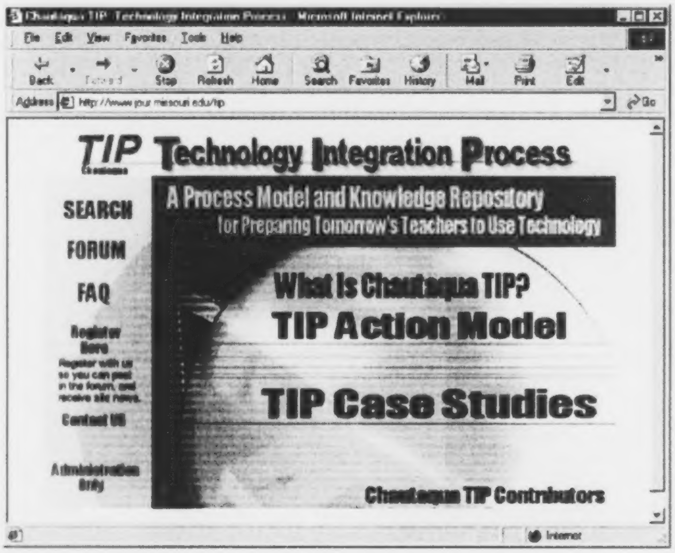

File Edit View Favorites Tools Help
Back
Forward
Stop
Refresh
Home
Search
Favorites
History
Mail
Print
Edit
Address
Go
TIP
Technology Integration Process
A Process Model and Knowledge Repository
for Preparing Tomorrow's Teachers to Use Technology
SEARCH
FORUM
FAQ
Register Here
Register with us so you can post in the forum, and receive site news.
Contact US
Administration Only
What Is Chautaqua TIP?
TIP Action Model
TIP Case Studies
Chautaqua TIP Contributors
Internet

Figure 3. The entry screen of the TIP system

To start with, the TIP design includes a process model for technology integration in teacher education. The model consists of five inter-related processes: research, design, development, delivery, and evaluation. Two or more subprocesses were identified for each process. This TIP action model represents the P component of Ravens' learning model: the theory of technology integration in teacher education. To capture the TIP experiences (the Q component) of participating programs, dedicated project staff was sent to collect knowledge about TIP actions in each participating program. These experiences categorized according to five TIP processes were written as descriptive documents enhanced by multimedia elements. Each experience is titled as a TIP case and can be searched by name, by category, and by program as well as through a full-text search engine spanning the entire database.

Following the action learning approach, knowledge collection at each participating program began with a two-day self-study facilitated by the project facilitator. The self-study process opened with an orientation to the goals of the project and a demonstration of the knowledge repository from the functional and conceptual perspectives. Attention then turned to identification of the strengths and limitations of the program in relationship to the elements of the TIP action model. More site visits followed for problem diagnosis and solution implementation. In these follow-up meetings, in light of presented problems with the program and illuminating TIP cases in the system, the participants reflected upon the problems and solutions to develop action plan. It was then up to the participants of each program to carry out the plan. In the meantime, project staff continued the tasks of TIP knowledge discovery and collection and preserved the knowledge in the system that also facilitates the information exchange and knowledge dissemination throughout project lifespan. Individuals from participating programs are periodically notified when new knowledge (i.e., documents) was added to the system. Also, notifications are sent out when a new threaded discussion is initiated or when existing discussions are active. In this way, TIP action learning becomes an ongoing and collective effort from all contributing partners of this consortium.

7. Conclusion

The utilization of the Web for action learning is one possible efficient and effective way to leverage the intellectual capital of an organization in solving organizational problems. In this paper, we have laid out the design framework of a Web-based action learning environment. We illuminate our design with an actual WALE that we built to integrate technology in teacher education. We also understand a successful WALE is more than a design framework and a new technology implementation. While Web technology may have the advantage of removing boundaries of space and time to facilitate and enhance action learning, it may cause other difficulties by eliminating ordinarily desirable interpersonal communication channels necessary for effective action learning. Our experience has shown that the successful application of a WALE relies upon a judicious marriage of a sound design of Web technological tools and resources and inner strengths of participants, with reflections upon learnings from experiences of action in the real world of work and life. It must focus on critical successful factors that include fostering a conducive learning culture, marshaling true leadership support, deploying a nurturing process model, and sustaining the change throughout the organization. Also, it must move us to a view that sees learning in the context of the workplace so that higher individual and organizational performance can be achieved.

References

[1] Lave, J. & Wenger, E. "Situated learning: Legitimate peripheral participation", New York: Cambridge University Press, (1991).

[2] Brown, J. S., Collins, A., & Duguid, P., "Situated cognition and the culture of learning. Educational Researcher", Vol. 17, pp. 32-42, (1989).

[3] Bonk, C. J. & Kim, K. A., "Extending sociocultural theory to adult learning", In Smith, M. C. and Pourchot, T. (Eds.), Adult learning and development: Perspectives from educational psychology, Mahwah, NJ : Lawrence Erlbaum Associates. (1998).

[4] Tharp, R., & Gallimore, R., "Rousing minds to life: Teaching, learning, and schooling in a social context", Cambridge, MA: Cambridge University Press, (1988).

[5] Wertsch, J. V. "Voices of the mind: A sociocultural approach to mediated action", cambridge, MA: Harvard University Press, (1991).

[6] McLaughlin & Thorpe. R., "Action Learning: A Paradigm in Emergence", British Journal of Management, Vol. 4., No. 1., (1993).

[7] Yorks, L., "The emergence of action learning", Training & Development, Vol. 54, No. 1, pp. 56-56. (2000).

[8] McGill, I. & Beaty, L. "Action learning: A guide for professional, management and educational development", London, UK: Kogan Page, 2nd Edition (1995).

[9] Revans, R., "Action learning: New techniques for action learning", London, UK: Blond and Briggs, (1983).

[10] Salomon, G., "Novel constructivist learning Environments and novel technologies: Some issues to be Concerned with. Research Dialogue in Learning and Instruction", Vol. 3, No. 1, pp. 3-12, (1998).

[11] Morris, K., "Wiring the ivory tower", Business Week, pp. 90-92. (1999, August 9).

[12] Revans, R., "The nature of action learning", Management Education and Development, Vol. 10, No. 1, pp. 3-23. (1979).

[13] Korey, G. and Bogorya. Y., "The managerial action learning concept-theory and application", Management Decision, Vol. 23, No. 2, pp. 3-11, (1985).

[14] Mumford, A., "Learning in action", Personnel Management, Vol. 23, No. 7, pp. 34-37, (1991).

[15] Beaty, L., Bourner, T., & Frost, P., "Action learning: reflections on becoming a set member", Management Education and Development, Vol. 24, No. 4, pp. 350-367, (1993).

[16] Marsick, V. J., "New paradigms for learning in the workplace", In V. J. Marsick (Ed.), Learning in the workplace (pp. 11-30). New York: Croom Helm, (1987).

[17] Senge, P. M., "The fifth discipline: The art and practice of the learning organization", New York: Doubleday/Currency, (1990).

[18] Froiland, P., "Action learning: Taming real problems in real time", Training, Vol. 31, No. 1, pp. 27-34, (1994).

[19] Keys, L., "Action learning: Executive development of choice for the 1990s", Journal of Management Development, Vol. 13, No. 8, pp. 50-56, (1994).

[20] McCarthy, R., "Multimedia: What the Excitement's All about", Electronic Learning, Vol. 8, No. 8, pp. 26-31, (1989).

[21] Harasim, L. M., "Global Networks: An Introduction", In Linda, L. M. (Ed.) Global Networks: Computers and International Communication. Cambridge, MA: The MIT Press, pp. 3-13, (1993).

[22] Kozma, R., "The Implications of Cognitive Psychology for Computer-based Learning Tools", Educational Technology, Vol. 27, pp. 20-25, (1987).

[23] Amthor, G., "Interactive Multimedia in Education: Concepts and Technology, Trends, and Model Applications, Megamedia and Knowledge Systems", T.H.E. Journal, Special Issue: IBM Multimedia, pp. 2-11, September, (1991).

[24] Jonassen, D. H., "Acquiring Structural Knowledge from Semantically Structured Hypermedia", A paper presented in the AECT annual convention, Washington, D.C. (1992).

Developing an Effective Web-Based Learning Environment for Overseas Chinese Education

Grace Chen and Kent Wang
Potomac Institute
47618 Loweland Terrace
Potomac Falls, VA 20165, USA
Grace@pidcusa.com

The world of education is in a period of rapid change. Technology in the education has recently become a primary goal of Overseas Chinese Education. Yet with all these new resources available to teachers, the opportunity for improved teaching is eclipsed by the intimidating task of finding ways to utilize computers and the Internet in a classroom environment. Overseas Chinese demographics are placing pressure on educational institutions to develop more cost-effective instructional delivery systems. In response to this pressure, education is exploring new ways of defining classrooms and utilizing distributed resources. The direction of this exploration is being guided by newly evolving technologies and information delivery systems, advances in neuropsychology and the cognitive sciences, and new philosophies and educational paradigms. The introduction of the Internet in Overseas Chinese Education has been the seminal event precipitating the emergence of one such paradigm characterized by fluidity of roles, individual learner-directed content, distributed resources, virtual facilities, and asynchronous class times. It uses technology to create learning environments with neither walls nor clocks. This paper will explore these technologies and how to develop an Effective Web-based instruction Learning Environment for Overseas Chinese Education.

Keywords: **WBI, Web-based Learning, Instruction Design, Overseas Chinese Education**

1 Introduction: The Need for WBI

As education explore new delivery systems and environments, it is necessary to observe and evaluate their effects on instructional quality and student learning. Some seek to shift the bulk of instruction to distance and distributed environments as a solution to the problems facing Overseas Chinese education today. Traditional, synchronous delivery methods utilizing physical facilities, teachers, and students have been through countless iterations and refinements. Their interactions and effects have been measured. The role of reading materials, visuals, lectures, feedback, demonstrations, and student dynamics has all been observed over many years. Before abandoning or reducing the traditional, it is necessary to ensure any modifications of existing paradigms will result in instruction that is at least as beneficial to both learners and educators as the existing one. Numerous reform movements and technical innovations have been introduced into classrooms in recent years. We have learned that these changes will not be accepted unless they are perceived to be beneficial for both teachers and students.

2 The Web Impact Instruction Design

Several things generally happen as teachers begin to use the Internet. The first occurs as teachers realize their pedagogical style needs to change if they intend to use the Internet significantly for teaching. This is typically a gradual shift but pedagogical styles do seem to change because of the robust nature of resources available and the difficulty of control over Internet usage. Some research on non-Internet network teaching activities also supports this idea. Generally, studies have found that when technology is introduced into the classroom, students experience an increase in motivation and self-esteem, accomplishment of more complex tasks, development greater technical skills and utilization of outside resources. Few studies, however, exist on the effects on student learning in distributed environments.

By combining the attributes of both Delivery of Content and System Management, and by answering in the positive the questions/issues unique to each, the instructional web-page developer will be more likely to create a strong and viable system/program that will teach, train, instruct, etc, all those whom they hope to educate via their instructional web sites. By adhering to these attributes, such a system can truly be called an Instructional System.

3 Creating good Web-based instruction

Not only does Web-based instruction need to follow good instructional design principles, but it needs to conform to good teaching practices and sound Web design principles as well. The first question educators should ask themselves before deciding to convert a course to the Web is, under the existing circumstances, is the Web itself an appropriate delivery medium.

Simply putting a course online because it is a new technology is not sufficient cause to justify the development time and cost. Another major concern is whether the online course will provide for the same level of quality teaching that a traditional class offers. Students will not accept the course if they perceive that it will be inferior.

Creating good Web-based instruction is not simply putting lecture notes online nor is it merely creating a virtual library of links to content related sites. It includes ensuring that good instructional, teaching, content, cognitive, visual, and usability design principles are followed as well as ensuring that it fulfills both teacher and student needs. Users need to perceive WBI as following objectives:

- **The Online Syllabus**
  An online syllabus provides the instructor with a way to change course material easily, and the student with a complete and up-to-date picture of the course requirements. The format need not duplicate the print version. Hypertext links to sample relevant disciplinary web sites may be helpful in giving students a sense of the disciplinary context for the course.

- **Personal Home Pages**
  Personal home pages can be used to foster the sense that the class is not just a collection of isolated individuals but a community of learners who can profit from interacting with one another. Home pages encourage students to learn about each other so as to encourage contact and mutual interests.

- **Interactivity**
  Adding discussion forums and chat sessions to your online course is a common way to add an interactive component to a web-based course. There are many implementations of bulletin board and chat session software to choose from. A second method of interactivity is, of course, e-mail. It' s a good practice to have an online list of the e-mail addresses of all students.

- **Assignments**
  The web page listings of homework assignments, upcoming events and exams can be more interactive than the familiar print counterparts. If some homework assignments, for example, are based on online materials, they can be directly linked to the class schedule.

- **Announcements**
  To be effective, announcements need to be read; for that to happen, students need to know when a new announcement has been posted. Alert sounds or perhaps a blinking link added to a page can let students know of new announcements, or perhaps, even a mass e-mail to all students in the course.

- **Testing**
  Online drill or practice testing can be used to reinforce material, even if the results are not used as part of a grade. Reading comprehension questions, for example, in short answer or multiple choice

formats can provide students with an assessment of their level of understanding of text.

- **Content**
  Perhaps the most difficult part of developing a web-based course is creating the online content. You can begin by transferring your basic lecture materials to the web and integrating media such as sound, images, and video. Remember, to experiment with incorporating some of the new web-based learning paradigms described above.

4 Summary

In conclusion, developing an effective Web-based learning environment for Overseas Chinese education is not simply putting lecture notes online nor is it merely creating a virtual library of links to content related sites. It includes ensuring that good instructional, teaching, content, cognitive, visual, and usability design principles are followed as well as ensuring that it fulfills both teacher and student needs.

Developing Web-Based Language Learning Environment

Yun Du
Second Language Teaching and Curriculum Center
Moore Hall 255, East-West Road
Honolulu, Hawaii 96822
Tel: (808) 956-5130
E-mail: ydu@hawaii.edu

The World Wide Web is becoming a popular media to conduct distance learning. However, using the Web for distance language learning is still a challenge. This paper introduces a Web-based language learning environment which is used to deliver upper-level second language courses. The three major design considerations (instructional design, interaction design and knowledge-building capability design) are discussed. The functions and major features of the learning environment are also described.

Keywords: **Web-based learning environment, computer-assisted language learning**

1 Introduction

Today's distance learning no longer assumes that knowledge is static and education is a certain years' procedure. Instead, knowledge is changing and evolving, so that education is a life long process. Education should be able to meet people's needs, no matter how old they are, where they are, or what jobs or positions they hold. The World Wide Web, with its worldwide access and friendly interface, becomes the desired media for conducting distance learning today. Compared with traditional classroom teaching, Web-based learning offers new opportunities:

- It extends the boundaries of learning so that learning can occur at any time, in any place. As a result, learners have more flexibility of choosing the way to learn.
- It emphasizes on collaboration and interaction that can be effectively employed toward learning. Using the Web, learners can not only communicate with the instructors or classmates, but also can go beyond the classroom to collaborate with people from other schools, institutions, organizations, and to ask questions to professionals and experts.
- Various resources of information on the Web extend the content of the instruction. Students are able to access multimedia information on almost every subject and in multiple languages.
- Web-based instruction offers opportunities for more creative activities. Students can search information on the Web, create their own resource repository, meet virtually with classmates and instructors, and do a lot more.

In this paper, a Web-based learning environment for language learning will be introduced. This learning environment is used for delivering instructional resources in Chinese at the 3rd/4th-year level and in Korean at the 2nd/3rd-year level to learners nationwide via the Web. The first Web-based class using the learning environment, CHN399 *Chinese Reading and Writing Course*, was officially offered to the students at University of Hawaii in the spring semester, year 2000. The last two units of the course also involved students from Taiwan to collaborate with students at UH. The course is a success. Twelve students have completed the course. In the fall semester, in addition to this course, a *Chinese Listening and Writing Course* and a *Korean Reading and Writing Course* will also be offered, using the same learning environment.

2 Major Design Considerations

The design of the Web-based language learning environment has been focused on three parts: instructional design, interaction design and knowledge-building capability design.

2.1 Instructional Design

Instructional design addresses the pedagogical issue of language learning. It determines the goal and format of the course, the instruction approach, the activities involved and the evaluation criteria. For this learning environment, we adopt an instructional model that contains the following five stages:

- Stage 1 Pre-activities: aim at activating students' prior knowledge, and helping them predict the content of the text
- Stage 2 Global activities: emphasize on helping students master the content and the main points of the text
- Stage 3 Specific information activities: use various language-centered tasks to train students to memorize the main points of the text while reading
- Stage 4 Linguistic activities: allow student to use the new knowledge after the they have mastered the content and main points of the text
- Stage 5 Post activities: integrate what have been learned from the previous four stages and help students accomplish a language task that involves using the new words, concepts and knowledge

Based on these stages, a series of activities have been designed. Students are evaluated mainly by the quality of their writings, and also by the quiz conducted at the end of each unit.

2.2 Interaction Design

Communication is very important for language learning. Communicative language learning theory emphasizes on interaction between learner and instructor as well as between learner and learner. Underwood [3] proposed a series of "premises for communicative CALL" (CALL refers to Computer-Assisted Language Learning), including "focuses more on using forms than on the forms themselves", "Teaches grammar implicitly rather than explicitly", "allows and encourages students to generate original utterances rather than just manipulate prefabricated language", etc. Interaction has been carefully designed to embed these principles into the learning environment. Different Web-based forums were developed for different learning purposes: asking questions about the text, practicing language through task-based group discussion, diagnosing grammar mistakes and writing essays on a specific subject. The asynchronous communication mode provides the following advantages:

- Students can have more flexible schedule; they can access the class at any time.
- Students have more time composing messages, and can modify messages even after they have been submitted. This is good for language learners who not only concern the content of the message but also the form of the language.
- Students can save specific messages for future reference.
- Students can search and retrieve specific messages afterwards.

2.3 Knowledge-building capability design

There are three major aspects of current learning theories. First, learning is a process of knowledge construction, not of knowledge recording or absorption. Effective learning depends on the intentions, self-monitoring, elaboration and representational constructions of the individual learner [2]. Second, learning is knowledge-dependent, and knowledge-driven [1]. People use current knowledge to construct new knowledge. Third, learning is highly tuned to the situation in which it takes place. Knowledge is not independent of the contexts (mental, physical, and social) in which it is used [2]. These theories indicate the importance of supporting knowledge building in a learning environment so that students can acquire, record, share, and integrate knowledge.

In our Web-based language learning environment, in addition to allowing students to discuss and share ideas in the forums, based on the characteristics of language learning, the knowledge building also includes following processes:

- Store resources related to the subject
- Build word vocabulary
- Compile grammar rules
- Collect and comment on writing examples, commonly used phrases and idioms, etc.

3 Major features of the Web-based language learning environment

The components of the final system are shown in Figure 1.

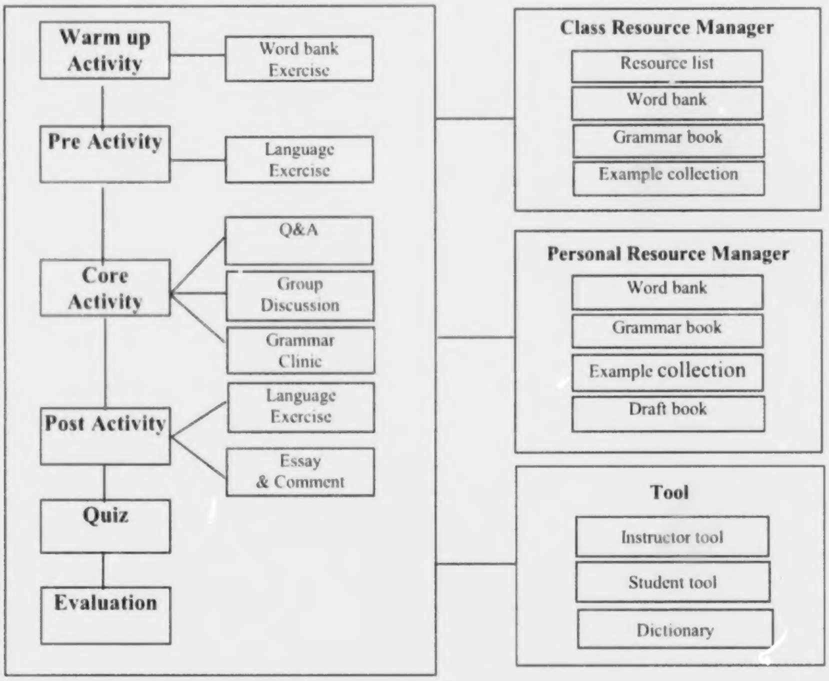

Warm up Activity
Word bank Exercise
Pre Activity
Language Exercise
Core Activity
Q&A
Group Discussion
Grammar Clinic
Post Activity
Language Exercise
Essay & Comment
Quiz
Evaluation
Class Resource Manager
Resource list
Word bank
Grammar book
Example collection
Personal Resource Manager
Word bank
Grammar book
Example collection
Draft book
Tool
Instructor tool
Student tool
Dictionary

Figure 1. Components of the Web-based language learning environment

3.1 Language teaching/learning support

The system supports language teaching based on a specific instructional model that sequences the learning process into several stages. The goals, processes, activities and tasks are well integrated into the functions of the system. Different rights and privileges are assigned to instructors and students to ensure that the teaching and learning procedure is followed. Using the system, the students are able to do language exercises, share information, ask questions, participate in task-based group activities, write essays, comment on fellow students' writing, build vocabulary, summarize grammar points, and so on, while the instructors are able to teach reading and writing skills, answer students' questions, correct grammar mistakes and evaluate students' progress.

Based on the instructional model, the learning process is sequenced into the following activities:

- Warm up activity: involves students' building word vocabulary (called *word bank*). This activity corresponds to the first stage of the instructional model: pre-activity that aims at activating students' prior knowledge about the topic.

- Pre-activity: involves doing language exercise such as matching words. This activity is also part of the first stage of the instructional model.

- Core activity: contains three parts. The first part involves students reading text (that is stored on CD-ROM), doing reading comprehension exercises and asking questions. This part corresponds to the second stage: global activities, and the third stage: specific information activities. The second part of the core activity is for students to participate in small group discussion to accomplish a given task, e.g. decide where to eat dinner. In the third part, instructors select mistakes from students' messages and post them in a forum called grammar clinic, and the students are asked to correct these mistakes. These two parts are designed to fulfill the goals of stage four: linguistic activities.

- Post activity: requires students to write an essay on the given topic. This activity is designed to integrate the knowledge they have learned, which corresponds to the fifth stage of the instructional model.

In addition to these activities, each unit of the class also has quiz, aiming at evaluating students' mastery of the material through quantitative criteria.

3.2 Database support

The system is developed using database technology. The database system is implemented on Microsoft SQL server. Basically the database system collects the data generated by the activities involved in the class.

- Stores the information of students and instructors
- Supports word bank
- Supports forums for class interaction
- Supports class and personal resource manager
- Supports quiz and grading
- Supports collecting survey data
- Collects data for administration such as login records

3.3 Asynchronous interaction

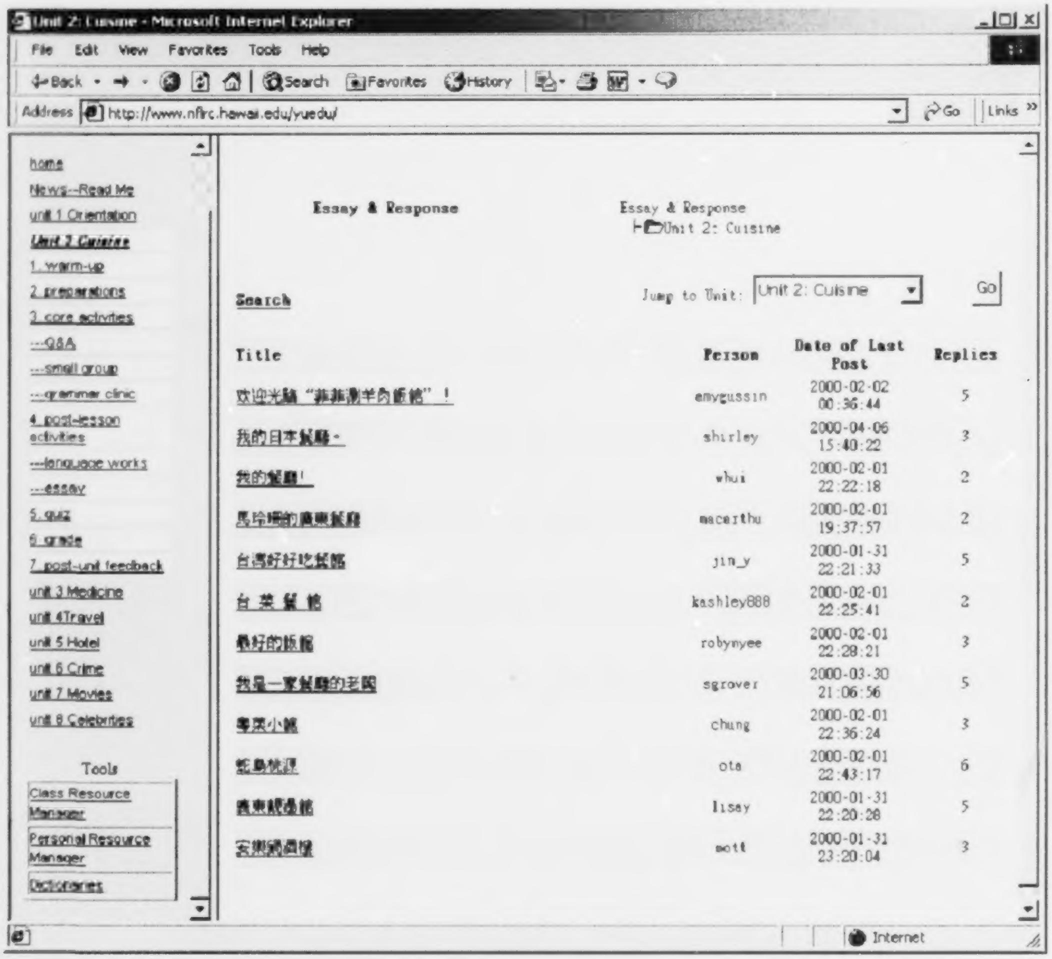
Web-based forums support the interaction among users. The asynchronous forums allow students to do the following things:

- To participate in the activity at any time
- To edit a message even after it has been submitted
- To save a specific message for future reference or as a knowledge item
- To search and to retrieve messages

The class interaction is supported by five forums:

- ***Class news forum***
  This forum is for instructor and students to exchange information including class announcement, cultural trivia, etc. Both instructors and students can post threads as well as replies.

- ***Essay forum***
  This forum is for students to post their essays and comment on essays written by fellow students. Both instructors and students can post threads as well as replies. (See Figure 2.)


Unit 2: Cuisine - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://www.nflrc.hawaii.edu/yuedu/
Go Links
home
News--Read Me
unit 1 Orientation
Unit 2 Cuisine
1. warm-up
2. preparations
3. core activities
---Q&A
---small group
---grammar clinic
4. post-lesson activities
---language works
---essay
5. quiz
6. grade
7. post-unit feedback
unit 3 Medicine
unit 4Travel
unit 5 Hotel
unit 6 Crime
unit 7 Movies
unit 8 Celebrities
Tools
Class Resource Manager
Personal Resource Manager
Dictionaries
Essay & Response
Essay & Response
Unit 2: Cuisine
Search
Jump to Unit: Unit 2: Cuisine
Go
Internet

Title	Person	Date of Last Post	Replies
欢迎光临“菲菲涮羊肉饭馆”！	amygussin	2000-02-02 00:36:44	5
我的日本餐廳。	shirley	2000-04-06 15:40:22	3
我的餐廳！	whui	2000-02-01 22:22:18	2
馬玲瑪的廣東餐廳	macarthu	2000-02-01 19:37:57	2
台灣好好吃餐館	jin_y	2000-01-31 22:21:33	5
台 菜 餐 館	kashley888	2000-02-01 22:25:41	2
最好的飯館	robynyee	2000-02-01 22:28:21	3
我是一家餐廳的老闆	sgrover	2000-03-30 21:06:56	5
粵菜小館	chung	2000-02-01 22:36:24	3
[illegible]島快評	ota	2000-02-01 22:43:17	6
廣東[illegible]館	lisay	2000-01-31 22:20:28	5
安樂[illegible]	mott	2000-01-31 23:20:04	3

Figure 2. Essay forum in the Web-based language learning environment

- ***Q&A forum***
  This forum is for students to post questions regarding the content of the text as well as the usage of the CD-ROM. Both instructors and students can post threads as well as replies.

- ***Small group discussion forum***
  This forum is for students to participate in task-based group discussion. Students will be directed into their group when they enter the forum and they can post messages there. They can go to see other groups' interaction, but they cannot post any messages in other groups' discussion area.

- ***Grammar clinic forum***
  Instructors select grammar mistakes from the students' posts, put them in this forum, and ask students to correct them. Only instructors can post threads, students can only post replies.

Designing individual forum for each activity or task makes it possible for forums to serve different purposes and to have different controls over students' privilege of posting messages. For example, in grammar clinic forum, only instructors can post threads (students can only post replies), but in class news forum, everybody can post threads. In all the forums, the instructors reserve the rights to delete messages.

3.4 Knowledge building

The system provides knowledge-building capability that allows users to gather information, discuss ideas with others as well as generating, storing and retrieving knowledge. The knowledge building process is facilitated using two tools: class resource manager and personal resource manager. Both the resource managers include resource list, word bank, grammar book, and example collection. In order to support knowledge building at both collective level and individual level, the knowledge- building tool has two types: one for the whole class (class resource manager), and one for the individual student (personal resource manager). Class resource manager can be accessed by the whole class, while personal resource manager is individualized and can only be accessed by student himself or herself. The personal resource manager also includes a draft book for the student to store his or her writing drafts. The instructors have most of the control over class level knowledge-building tool, but the students have full control over their own knowledge-building tools. The knowledge-building functions provided include:

- Allows knowledge-building at both class level and individual level
- Allows users to collect information resources (Web sites, article, etc.) into *resource list*
- Allows users to collect words into *word bank*
- Allows users to collect or compile grammar rules into *grammar book*
- Allows users to collect writing examples or idioms into *example collection*
- Allows users to save messages from discussion forums to the resource managers
- Allows users to write note or comment on resource or knowledge items

Currently, the grammar book and the example collection in class resource manager are controlled by the instructors, meaning that instructors summarize grammar points and select examples and put them in the class resource manager. Students can read them but they cannot put their own notes there.

3.5 Online quiz

Students can take quizzes online. The quiz contains multiple choices and is graded automatically so students can get their grade immediately after they submit the answer. Instructors can check students' quiz grades along with the information such as how long the students complete the quiz and their answers to each question.

3.6 Tracking capability

The tracking system built for this learning system record student's clicks into the database while they are navigating the class. The information recorded include the location of the student, the action the student makes, the time of the action and other relevant data such as the message the student is reading. The tracking system offers the following benefits:

- The tracking system can provide valuable information for system developer. Users use a system in different ways. Therefore, how users navigate our the learning system, how they use the interface and how they use the various functions become interesting questions whose answers will help the system developer understand the operation of the system so that the system can be improved to better meet users' needs.

- The tracking system also provides valuable information for the instructors. It tells the instructor how students do self-learning in the distant environment, how they follow the process designed for the course, how they participate in the activities, how they approach a task, how they respond to a teaching strategy, and so on. The information will help the instructors understand students' behavior (e.g. learning strategy) and adjust their teaching methods to make the course more effective.

3.7 Monitoring and evaluating performance

The system provides ways for the instructors to conduct teaching as well as monitoring and evaluating students' performance:

- Monitor students' participations according to their login records, frequency and length of their posts, and so forth
- Evaluate students' performance according to their participations, contents and form of their writings, and so forth
- Give grades and feedbacks to students
- Understand students' learning behavior by analyzing tracking records

4 Conclusions

Observations from the Web-based Chinese reading course show that this Web-based language learning environment successfully support the class operation. Students and instructors are able to choose their own time, place and pace to work on the course. And, they have been engaged in active interaction during the course. The functions provided by the learning environment meet the instructional goals and requirements.

References:

[1] Glaser, R., "Education and thinking: The role of knowledge", American Psychologist, Vol 39, pp. 93-104, (1984).
[2] Resnick, L. B., "*Knowing, learning and instruction*", Hillsdale, NJ: Lawrence Erlbaum Associates, (1989).
[3] Underwood, J., "Linguistic, computers, and the language teacher", Rowley, MA: Newbury House, (1984).

Development of 3D simulation programs for classical mechanics - Using Java 3D -

Yasuyuki NAKAMURA* and Hiroshi NAKANO**
School of Informatics and Sciences, Nagoya University
Nagoya, 464-8601, Japan
*Tel: *+81-52-789-4753 **+81-52-789-4801*
*E-mail: *yasuyuki@info.human.nagoya-u.ac.jp*
***nakano@info.human.nagoya-u.ac.jp*

1 Introduction

As LAN and Internet have diffused in recent years, environments of computers have been well filled and computers have become more popular among students. These developments make it possible for the style of education of physics to become various.

Recently many groups develop simulation programs for education of physics. We develop simulation programs for physics education using C language, XForms and Mesa library on Linux computer [1]. There are not only 2D but also 3D simulation programs. These programs are used in physics experiments for students [2,3]. One of the weak points of our system is a difficulty for opening our programs to the public. Only students attending the class can execute the programs.

On the other hand many programs coded by Java are also proposed and an education environment are prepared on the Internet. On World Wide Web (WWW), there are many programs coded by Java that are opened to the public [4,5]. Condensed Matter Theory Group of Kyushu University in Japan opens a virtual laboratory for the introduction of physics on WWW [6]. In the laboratory one can study physics with a simulation and an explanation for it. It is very good idea to open the programs to the public through Internet, but most of those programs are for 2D simulation. We think that 3D simulation is more exciting and is more helpful for understanding the motion of objects, because 3D simulation is more realistic.

Our aim of this study is to develop 3D simulation programs and open them to the public through Internet. We develop programs using Java in order for as many as people to utilize them and make use of Java 3D API for realizing 3D visualization. To our knowledge, there are still only a few programs proposed for education with use of Java.

2 Environment of development and execution

Our simulation programs are developed on an IBM PC/AT compatible computer. We adopt Linux as an operating system (OS) and XFree86 3.3.6 for X Window system. Java 2 SDK v 1.2.2 for Linux Production Release, Java 3D 1.1.3 API and Mesa 3.1 are used for developing applications. One of the reasons for adopting Linux is that Linux system has a reputation for its stability. Although applications are developed on a computer with Linux operating system, one can use any kind of computers and operating systems for execution of applications. This is merit and what we aim for in developing applications.

3 Example

We developed some programs with use of Java and Java3D by way of trial. One of them is a simulation of motions of ball in a box under gravitation, whose snapshot is shown in figure 1. We list below special features of these applications.

- We make use of Swing API for graphical user interface (GUI). Swing is provided as one of the standard APIs in Java2 and we can develop applications with common GUI operations in total independence of a kind of a computer and OS using Swing API.
- One can execute programs not only as an application on a local computer but also as an applet on a

browser through Internet. Java Plug-in is necessary for executing programs on a browser in the present. However this restriction will be solved in the future.

- Real-time simulations can be realized with use of thread class of Java. Furthermore, one can execute applications in slow-motion mode and in fast-forwarding mode.
- Java3D uses a tree structure for realizing 3D visualization. By changing branches and leaves, objects can be moved, transformed, replaced and so on.
- Java is a class-based object-oriented programming language. Therefore we can easily add or remove objects. Furthermore rules of motion can be specified to objects. Then we can realize various motions of objects.
- Since Java2 is prepared for Unicode and Locale, internationalized programs can be developed.

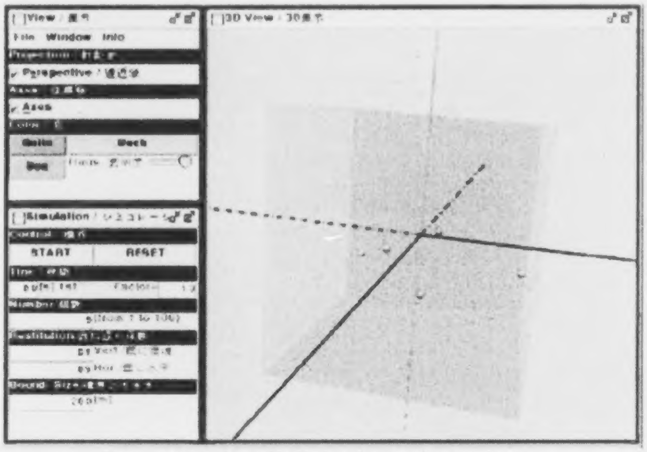


Figure 1. Motion of balls in a box under gravitation.

4 Conclusions

We propose an educational system for elementary physics with use of Java and Java 3D API. Our system offers 3D simulation programs with use of Java 3D. 3D visualization of the system of classical mechanics helps students to understand the behavior of the system and to have interests in physics. Since our programs are developed by Java, anyone who has an environment of Java can execute them on WWW. Therefore we can open our programs to the public and we can receive responses and evaluations for our programs. Note that one needs Java Plug-in for execution of our programs in the present. In the future, we want to increase the number of simulation programs and open those programs to the public.

This work is supported by the Japanese Grant-in-Aid for Science Research Fund from the Ministry of Education, Science and Culture.

References

[1] Yasuyuki Nakamura and Hiroshi Nakano, "Development of simulation programs for classical mechanics –Using UNIX system –", Computer & Education, Vol. 6, pp. 107-111, (1999).

[2] Butsurigaku Kiso III Home Page, http://kiso3x0.phys.human.nagoya-u.ac.jp/ (accessed on August 9, 2000)

[3] Yasuyuki Nakamura and Hiroshi Nakano, "A Computer Network System for Physics Education –Using Linux –", Proceedings of the Seventh International Conference on Computers in Education (ICCE' 99), Chiba, Japan, November 4-7, pp. 821-822, (1999)

[4] JAVA Physics Resource Mailing List, http://www.nep.chubu.ac.jp/~nepjava/applet.htm (accessed on August 9, 2000)

[5] Simulations on Physics, http://www2.tky.3web.ne.jp/~inaga/index.html (accessed on August 9, 2000)

[6] Virtual Laboratory, http://www.cmt.phys.kyushu-u.ac.jp/virtuallab.html (accessed on August 9, 2000)

Development of a Web System to Support Computer Exercises and its Operation

Hajime Kitakami, Sueda Takashi, Yasuma Mori and Susumu Kuroki
Faculty of Information Sciences Research, Hiroshima City University
3-4-1 Ozuka-Higashi, Asa-Minami-Ku, Hiroshima 731-3194, JAPAN
Tel: +81-(0)82-830-1587
Fax: +81-(0)82-830-1587
E-mail: kitakami@its.hiroshima-cu.ac.jp

This paper describes the development and operation of a Web-based system to support computer exercises used in a course on data structures and algorithms. To develop such a system, this paper proposes using the functions of a Web-based system to deal with a learner' s state transition model based on computer exercises. The Web system developed by us has useful functions, some of which are the management of participant registration, identification of learner's goals, web service of exercises, mutual interaction between participant and teacher, management of report submissions, and both provision and analysis of electronic questionnaires to participants. The use of this system resulted in students' heightened motivation to work, good communication between participants and teachers, and a reduced workload for teachers.

Keywords: **Web, Database, Exercise, Autonomous Learning, Domain Model, Communication, Questionnaire, Data Mining, Operation, Evaluation**

1 Introduction

The new curriculum of the Department of Intelligent Systems at Hiroshima City University has added computer exercises to subjects related to algorithms and programming, thus encouraging students, from freshmen to sophomores, to make the most of their ability for practical programming with representative algorithms. The curriculum offers two ongoing three-hour courses that include theory and practice.

This paper focuses on computer exercises for the course "Data Structures and Algorithms," which is a part of the core curriculum for sophomore students. The general objective of the course [1, 2] is to facilitate the transition from computer literacy to a professional level of information processing. Even though students have considerable knowledge of computer operations, they do not have perfect command of them Moreover, they do not have enough experience in basic programming techniques. In order for them to have command of the theory and the practice, we have developed many exercises to improve the management of participant registration and learner's goals, information about the exercises, mutual interaction between participants and teachers, management of report submissions, and collection of questionnaires, among others. However, a problem arises because the workload for both teacher and students increases in the process. To solve this problem, we have developed the necessary support Web system dealing with a learner's state transition model based on computer exercises. Moreover, we report the operational results obtained from real exercises.

2 Assessment of learners' situation before the training

The contents of the courses "Data Structures and Algorithms I" and "Data Structures and Algorithms II" were divided into two courses, each including both theory and practice, using C in the new curriculum The former includes major elements such as stack, queue, list, naive sort, recursive function, quick sort, tree structure, and binary sort in the second semester of the first year. The latter includes major elements [3, 4] such as complexity, file processing, linear search, binary search, hash, B-tree, pattern matching, graphical

BEST COPY AVAILABLE

searches, Kruskal, and Dijkstra in the first semester of the second year. Since students can easily understand the content of many classes if they have attended C in an earlier semester, "Structured Programming" was also organized into two courses including both theory and practice using C in the first semester of the first year. This course includes major expressions such as if-, while-, and for-statements, array, data types, pointer, function, and structure in C. Moreover, the teaching of computer literacy includes major elements such as word processors (e.g., LaTex), programming tools (e.g., mule, e-macs), drawing tools (e.g., TGIF), the input tool for Japanese characters, electronic mail, X-window, and the shell command on UNIX, among others, in the same semester.

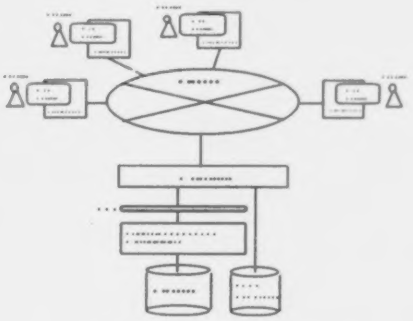


Figure 1. System Configuration

An evaluation of the learners' situation before starting the course "Data Structures and Algorithms II" that is the focus of this paper provided the following results:

(1) Students did not have much knowledge about algorithms and data structures with practical usage. They had learned simple and short programs but did not have much experience with longer programs. For example, they did not have experience in how to update longer programs by themselves.

(2) They did not have enough motivation for autonomous learning. They were less eager to learn than freshmen. For example, they did not consult textbooks or dictionaries on their own when they had trouble understanding an exercise.

(3) Twenty-five percent of the students did not understand the C language. Seventy-five percent of the students tended to forget the C language, since they had not had a chance to practice it for more than 2 months after the second semester of the first year.

(4) Many students did not have sufficient skills to attain perfect command of software tools such as TGIF or LaTex.

3 Conceptual view of the computer exercise

Figure 1 shows the system configuration to support the exercise. Since each learner does his exercises at a workstation connected to the Internet, he can access information managed by the Web server. The Web server stores the exercises as HTML documents. The application program located in the CGI (Common Gateway Interface) manages information related to his registration, personal goals, and questionnaires. The application program is implemented in Parl, Shell, and SQL. The information inputted by the Web browsers is stored in the database and used by the learners.

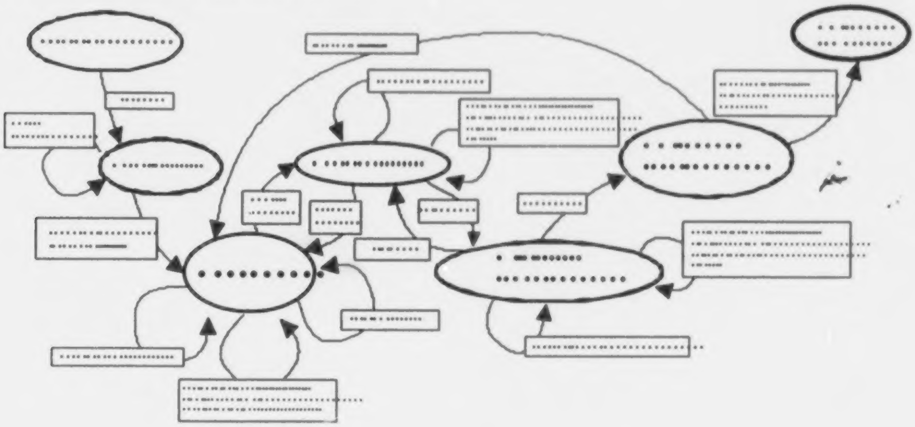


Figure 2. Computer Exercise Model

We tried to computerize human work as much as possible in the existing computer exercise. Notice of all 15 exercises included in the course was given on the Web page. We connected both basic programs and measurement data to the Web page for each exercise. Using a Web browser, both could be downloaded from the Web server to a student's site. Before starting on the first exercise, students had to fill out an electronic registration form for the class using the Web browser. When a student inputted his school number, name, password, and e-mail (electronic mail) address in the registration form, the system issued him a registration number using e-mail and the Web page. If the student needed any information about the exercises after that, he could get it by inputting his registration number and password using the Web browser.

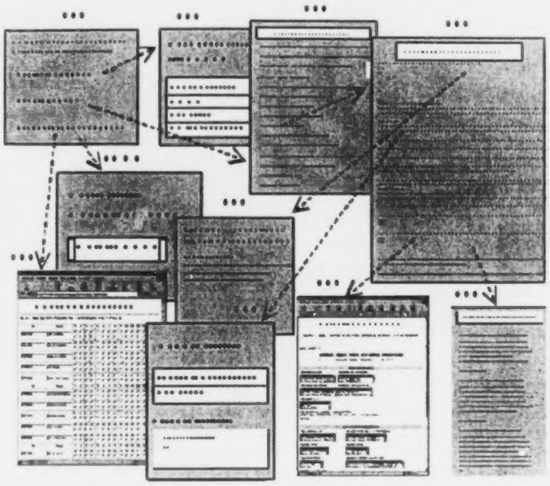


Figure 3. An Example of System Operation

Figure 2 represents the state transition for the computer exercise model. "Starting the Course," located at the left side of Figure 2, represents the state before starting the class. The student moves to the state of "Completing the Course" if he finishes all exercises successfully. If the student inputs personal data in the class registration form, the student moves to the state of "Class Participant." If the participant replies to the first questionnaire and inputs his personal goals for the exercise using the Web browser, he moves on to the state of "Exercising." At this stage, the learner is allowed to solve the exercise. If the learner inputs a question to the teacher on the Web page, he receives a reply from the teacher on the Web page. After finishing the exercise, the learner moves on to the state of "Making the Report" and can answer our questionnaire for the exercise as he finishes the exercise. If the learner submits his report to the teacher, he moves on to the state of "Waiting for the Evaluation." If the evaluation is poor, the teacher contacts the student, helps him, and asks him to re-submit the exercise. The Web system does not support their interaction in the situation, since we believe that face-to-face communication is preferable. This situation is different form Fujimoto's Classroom Management System [5]. After the learner reaches the state of "Completing the Submission," he will input his personal goals for the next exercise. After that, he will move to the state of " Exercising."

We place great importance on the use of educational methods [6, 7] including "Reading, Writing, and Using an Abacus" to achieve the goal of "autonomous learning and thinking." For students belonging to the categories (1)-(4) mentioned above, the computer exercise model includes the following educational methods. Students in (1) and (3) are asked to read longer programs downloaded from the Web server, write the respective flowchart, update the subparts, and measure their performance in the state of "Exercising" shown in Figure 2.

Students in (2) are asked to define their personal goals before reaching the state of "Exercising" and write a self-evaluation in the state of "Making the Report." In the state of "Exercising," students are given an ambiguous exercise to learn the value of searching for information. In this way, students are encouraged to develop their creativity skills. Moreover, students are strongly advised to use textbooks and dictionaries if they have unresolved questions. Students in (4) are strongly encouraged to use such tools as TGIF and LaTex when preparing a report that includes figures and text. We believe that longer programs particularly enhance their proficiency in using tools. In order to determine an accurate grade for each exercise, we evaluate the reports submitted by the students and their answers to the questionnaires. Since we receive the results of the questionnaires immediately through the Web, we use such results to improve the exercises and coach the students. Moreover, the students can also receive their scores in a very short time. Students can compare each other's scores if they are given access to the statistics. Giving students access to the statistics is regarded as the key to ensuring an environment of awareness [8].

BEST COPY AVAILABLE

4 The results of system operation

Figure 3 represents an example of the operation of the system. Web page number (1) in the figure relates to the state of "Starting the Course." Page (2) is the class registration form. Page (3) gives anchors for information about all 15 exercises included in the course. If a learner selects one of the exercises on the page, he can use the exercise page (4). He can access his record of submissions and re-submissions using Web page (5). After inputting his personal goals using Web page (6), he moves on to the state of "Exercising." When he finishes the exercise, he moves on to the state of "Making the Report" and inputs the questionnaire on Web page (7). The results of the questionnaires are immediately stored in the database. Not only the teacher but also the learners are able to compute the statistics of the results from the database in real time. Page (8) relates to the statistics. Pages (9) and (10) are for teachers' use only. In page (9), each student has 15 check boxes, each divided into an upper and a lower section. If the report evaluation is good in the state of "Waiting for the Evaluation," the teacher puts a checkmark in the upper check box. If not, he puts the checkmark in the lower check box and helps the student so that he re-submits his work. Page (10) is useful for analyzing questionnaires stored in the database. The analysis includes the method of data mining [9] implemented in SQL.

Application of the system operation started at the Department of Hiroshima City University in April 1999. This system motivates students to do their exercises, provides good communication between participants and teachers, and reduces teachers' workload. The evaluation results of questionnaires and examinations related to the exercises are as follows:

(1) Ninety percent of students studied for 0.5-2.0 hours at their homes and were interested in the lecture.
(2) Twenty-six percent of students spent less than 2.0 hours preparing the report and exercising, 53% spent 2.0-5.0 hours, and 21% spent more than 5.0 hours, not including class work.
(3) Seventy percent of the 12 students (25%) previously mentioned understood the C language. Moreover, all students made progress in their studies.
(4) Ninety-five percent of the students reported good understanding of the algorithms used in the exercises. Eighty-seven percent of the students passed the examinations.
(5) The students acquired good skills at using TGIF, LaTex, and other programs to write reports.
(6) Seventy percent of the students felt that the teacher did his best in the classroom, and 17% of them barely approved of his performance.

5 Conclusions

We proposed a computer exercise model for the course of "Data Structures and Algorithms II" and developed a Web support system for computer exercises using the model. We place great importance on educational methods including "Reading, Writing, and Using an Abacus" so that our students acquire the skills of "autonomous learning and thinking." Computer exercises using the Web system give students a chance to enhance their capabilities of "autonomous learning and thinking" and "creativity." The system run on the Web server has useful functions, some of which are the management of participant registration, identification of learner's goals, web service of exercises, mutual interaction between participants and teachers, management of report submissions, and both provision and analysis of electronic questionnaires to participants. The use of the system resulted in students' motivation to do the exercises, good communication between participants and teachers, and a reduction of teachers' workload. In order to achieve more concrete results, the students studied more at home and were enthusiastic about doing their exercises. Moreover, the students learned how to make a report using TGIF, LaTex, and other programs.

Acknowledgements

We thank Assistant Professor Dr. Akira Sato of Hiroshima City University for his help with the exercises. We also thank Mr. Jun Ogawa, Mr. Takashi Maesako, and Mr. Kohei Matsumoto, TAs (Teaching Assistants) of Hiroshima City University, for their help with the exercises. This work was supported in part by a Grant-in-Aid of Scientific Research (C) from the Japan Society for the Promotion of Science and a Hiroshima City University Grant for Special Academic Research.

References

[1] Denning P. J., Comer D. E., Gries D., Mulder M. C., Turker A., Turner A. J., and Young P. R.: Report of the ACM Task Force on the Core of Computer Science, Association for Computing Machinery Press (1988).

[2] Computer Science Educational Curriculum for B. A. Degree Program in Computer and Engineering J97, Information Processing Society of Japan, 158p. (1997).

[3] Aho A. V., Hopcroft J. E., and Ullman J. D.: Data Structures and Algorithms, Addison-Wesley (1983).

[4] Leendert A.: Programs and Data Structure in C, Second Edition, John Wiley & Sons (1992).

[5] Fujimoto R., Suganuma A., and Mine T.: Development of a Classroom Management System on the Web, Advanced Research in Computers and Communications in Education (ICCE'99), Ed. by Cumming G. et al., Vol. 55, IOS Press, pp. 756-759 (1999).

[6] Angelo T. A.: Beginning the Dialogue: Thoughts on Promoting Critical Thinking, Teaching of Psychology, Vol. 22, No.1, pp. 6-7 (1995).

[7] Kuhar M. B: Critical Thinking: A Framework for Problem Solving in the Occupational Setting, AAOHN JOURNAL, Vol. 46, No.2, pp.80-81 (1998).

[8] Dourish P. and Bly S.: Portholes: Supporting Awareness in a Distributed Work Group, Proc. of CHI'92, pp.541-547 (1992).

[9] Fayyad Usama M., Piatestky-Shapiro G., Smyth P., and Uthurusamy R.: Advances in Knowledge Discovery and Data Mining, AAAI Press/ MIT Press (1996).

Development of CAI System with Character Code Discrimination on WWW Environment

Hiroyuki Watanabe and Katsuhiro Kato
Department of Electrical and Electronic Engineering, College of Engineering,
Nihon University
1 Nakagawara, Tokusada, Tamuramachi, Koriyama, Fukushima 963-8642, Japan
E-mail: {hiroyuki, kkatoh}@ee.ce.nihon-u.ac.jp

1 Introduction

The CAI systems on the Worldwide Web are accessed by learners all over the world. However, the server-client type CAI system has a problem in that the character code does not translate into other character codes. Therefore, in the previous CAI system[1], the S-P chart used for data analysis was readable only in the Japanese version[2]. The new client program runs Java applet corresponding to the character code of the learner's language and the character code in the tag is transferred to the server together with the learner's data. The character code in the tag is decoded on the server side, and the HTML file provides the S-P chart. As a result, even if the CAI system is accessed from various countries, the character code of the learner's language, such as Japanese or English, can be decoded by one server program and the SP chart corresponding to the character code can be provided.

2 Flowchart for the Character Code Decoding

This CAI system is constructed through the WWW client program with Java applet corresponding to the character code of the learner's language, and the WWW server program with the Java application[1]. Below is a description of the process. (see Fig. 1).

(1) The Japanese or English learner selects Java applet in Japanese or in English, respectively. The questions or hints are displayed. The learner's answer is judged via the WWW client program which is online.

(2) When the WWW client is only one Java applet, the WWW server has a character code error for the difference between the languages of the client and the server. For character code decoding the following code is added in the tag by Java applet.
<GET> M dir_name learner_name JPN
..for Japanese
<GET> M dir_name learner_name ENG
..for English

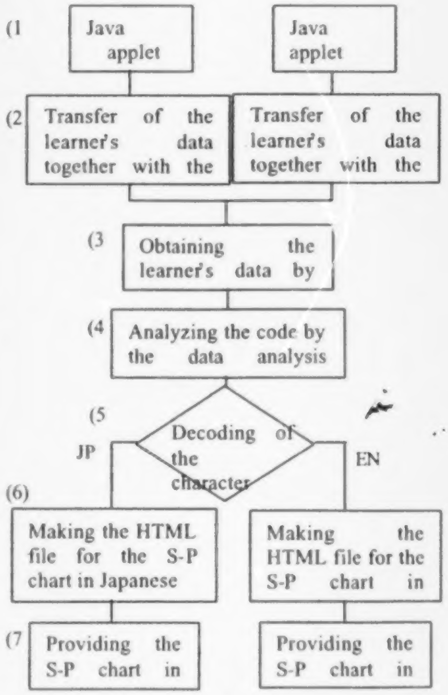

(1
Java applet
Java applet
(2
Transfer of the learner's data together with the
Transfer of the learner's data together with the
(3
Obtaining the learner's data by
(4
Analyzing the code by the data analysis
(5
Decoding of the character
JP
EN
(6)
Making the HTML file for the S-P chart in Japanese
Making the HTML file for the S-P chart in
(7
Providing the S-P chart in
Providing the S-P chart in

Fig. 1: Flowchart for the character code

The "<GET>" is one of the tags transferred from the client to the server. The code "M" is the data management related to the language. The "dir_name" is the directory name of the courseware for the saving of the learner's answers. The "learner_name" is the learner's name. The last code "JPN" or "ENG" is the character code of each learner's language.

(3) The character code together with the learner's data, which includes the learning score and its time, are obtained by the WWW server program through the Internet or the Intranet.

(4) Even if the language code is different, the learner's data is saved with the same file name in the same directory for the courseware. The learner's data is managed collectively, and the data analysis program analyzes the character code in the tag.

(5) The character code difference between Japanese and English is decoded.

(6) The learner's ranking is placed with the data of all learners, which has been stored in the server for each courseware. The S-P chart and the result of the statistical analysis which is formatted by the HTML are made corresponding to the character code of each learner's language.

(7) The S-P chart with the character code of each learner's language is provided to the WWW client.

3 Results

Fig. 2 shows the S-P chart in English for the score. The score for each learner is sorted vertically to the smallest value, which is the S-curve, and the score for each question is sorted horizontally to the smallest value, which is the P-curve. The S-P chart displays the learner's ranking. The attention coefficients for each learner and each question are shown. Furthermore, the evaluation of the learner, the average and its standard deviation are also shown. The S-P chart can also be accessed and displayed in Japanese.

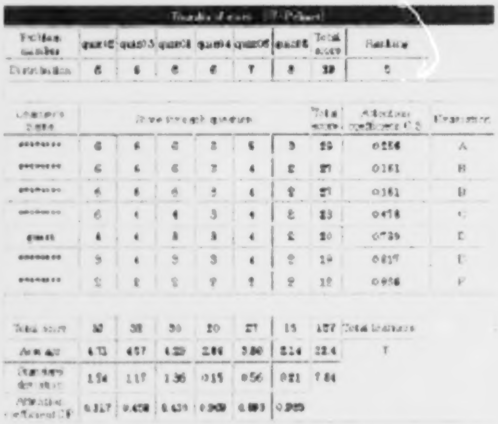


Fig. 2: S-P chart in English.

4 Conclusions

For a good study, it is important that the S-P chart be provided for the learner. In this paper, the character code corresponding to Japanese or English, together with the learner's data, is transferred to the server from the client by Java applet. The S-P chart, which is written in Japanese or in English, could be provided by one server program making the HTML file corresponding to the character code of the learner's language. As a result, many server programs will not need to prepare character codes for the learner's language. This should increase the number of learners and give learners more definitive rankings. Each learner can access the courseware by typing the following URL through the Internet.
http://133.43.15.87/~webcai/index_e.html

References

[1] H. Watanabe & K. Kato, "Development of the ICAI system in WWW for individualized education", Proceedings of the 5th International Conference for Young Computer Scientists (ICYCS'99), Nanjing, China, Aug.17-20, pp.966-970, (1999).

[2] H. Watanabe & K. Kato, "An analysis of response time distribution for CAI courseware", Proceedings of the 7th International Conference on Computer in Education (ICCE' 99), Chiba, Japan, Nov.4-7, pp.125-132, (1999).

Development of Intelligent Learning Support System with Large Knowledge Base

Katsuyuki SUGAYA and Setsuo OHSUGA
Department of Information and Computer Science, Waseda University
3-4-1 Ohkubo, Shinjuku-ku, Tokyo 169-8555, JAPAN
E-mail: {sugaya, ohsuga}@ohsuga.info.waseda.ac.jp

The objective of this paper is to present framework for developing intelligent learning support system with large knowledge base. Recently, the need for effective learning support and training is mounting, especially in industry or engineering fields, which demand the learning of complex tasks and expertise knowledge. Intelligent learning support system is being employed for this purpose, thus creating a need for cost-effective means of developing learning support systems. In this study, intelligent learning support system is assumed as a part of the intelligent knowledge management support system. The factors necessary for the intelligent learning support system discussed here are generality and adaptability. In order to achieve the goal, a new design of the system and learner modeling technique are discussed as well as a way of generating specific intelligent learning support system.

Keywords: **Intelligent System Design, Large Knowledge Base, Learner Model, Model-based Diagnosis, Knowledge Management**

1 Introduction

The purpose of this paper is to introduce a new framework for developing intelligent learning support system using large knowledge base. This system is a part of the intelligent systems that is developing to enable the expertise knowledge management.

In daily life, human has to interact with and reason about a large number of systems. This includes physical devices as well as non-physical systems. Also in professional work a growing number of people has to be trained in operating and designing large complex systems such as airplanes, nuclear power plants, and enterprises. Consequently, the goal of education or teaching may vary from inducing insight in the physical principles underlying the behavior of the device to teaching behavior analysis in the context of system design, operation, and maintenance. In addition, recently systems in the real world are becoming larger and more complicated. Rapid progress in science and technology has created a need for people who can solve complex problems and operate and maintain sophisticated equipment. In these situations, we, human beings, have to solve various types of problems using expertise in the large and complicated systems. Therefore the need for effective learning support or training is rising, given the increasing complexity of the workspace, especially in engineering or industrial fields.

Many computer assisted instruction techniques exist that can present instruction, and interact with students in a tutor-like fashion, individually, or in small groups [3]. The introduction of artificial intelligence technology and expert systems technology to computer assisted instruction systems gave rise to intelligent tutoring systems. In the intelligent tutoring system, for example, intelligent tutors that can model the learner's understanding of a topic and adapt the instruction accordingly [2]. Although intelligent tutoring systems research has been carried out for over 15 years, few tutoring systems have made the transition to the commercial market. Authors consider that some serious problems exist in the current methodology of developing intelligent tutoring systems. As an example, each system is developed independently, and tutoring expertise is hard-coded into individual systems. In particular, the problem of learner modeling technique exists as a basic issue. The system must have learner model that represents an estimate of the learner current understanding of the domain knowledge to be used by tutor in order to give adaptive guidance and explanations to the learner. A number of learner modeling techniques have been developed [8]. However, not every model can be called complete expressing the learning condition of the learner. Hence, the motivation for this study comes from the need for effective intelligent tutoring systems, particularly development of more complete learner modeling technique.

For these problems like above we consider that the factors necessary for the intelligent learning support system discussed here are generality and adaptability. In order to achieve the goal, authors present a new framework of the intelligent learning support system those enough practical conditions. Several concepts are included in this study; expert knowledge management with large knowledge base, knowledge sharing, knowledge processing, model-based learner diagnosis, etc.

2 Expert Knowledge Management using Knowledge Base System

In this section, we introduce briefly the key concept of our knowledge base system. Our research groups have tried to solve various problems by knowledge-centered intelligent system. The main concept is Multi-strata modelling scheme [5]. This modelling scheme is applied many intelligent systems, and these systems rewarded with good results, e.g. automatic programming system [1]. And we considered that Multi-strata model is strongly support the development of intelligent tutoring systems [6][7].

2.1 Intelligent System with Large Knowledge Base

At first, we discuss to apply large knowledge base for the architecture of intelligent learning support systems, which can generate learning support systems for a wide range of domain.

In these days, with the developing of science and technology, the systems which human manages with are enlarged and more complicated. In particular, it is too difficult to transmit expert knowledge from expert engineer to novice engineers. In the engineering field, even a large system developed by many expert engineers. When the system grows lager and more complex, the knowledge that is needed to build the system is more specialized and subdivided. In these situations, some serious problems are occurred. For instance, it is difficult to communicate between expert engineer and another fields' engineers or novice one. In other words, it is too more expertise to transmission of expert knowledge from human to humans. For this reason, the expert knowledge hiding is occurred in some engineering companies.

When the knowledge is specified and subdivided, in the situation like classroom, it is not appropriate to transmit the knowledge from expert engineer to novice one e.g. next generation engineers. Therefore, we propose the transmission of expert knowledge through the large knowledge base system (Fig.1).

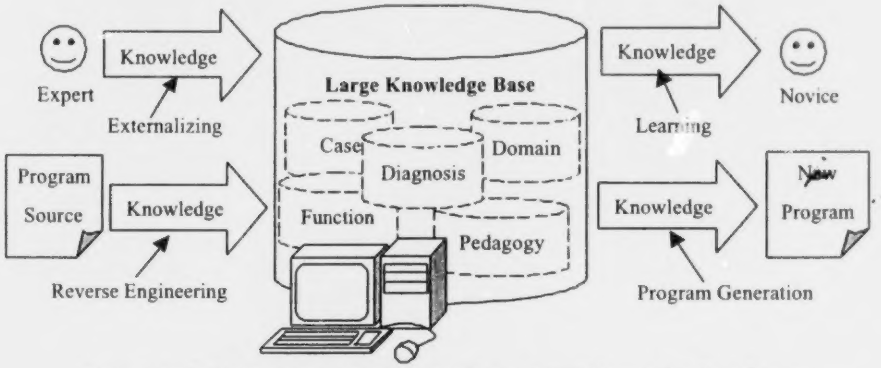


Fig.1: Knowledge Transmission using Intelligent Large Knowledge Base System

In this study, we consider that intelligent learning support system is a part of intelligent knowledge management support system. Moreover, we believe that knowledge management or learning support system is one of large and complex problem solving systems. The term problem is used here in a wide sense to mean what a person wishes to know or wants to do. There are various types of problems such as analysis,

BEST COPY AVAILABLE

design control, decision-making, planning, and teaching. Most of them are not well dealt with by conventional software method but require the system a capability to find a solution itself in a large space. Since the space is open, self-controlled exploration in the space is necessary. The system must be provided with the various methods to solve the different type of problems, each of which is represented by a specific knowledge chunk. Furthermore, a complex problem concerns different problem domains and since a problem requires domain specific knowledge, the system must be provided with a global knowledge base including the various type of domain knowledge.

In order to use knowledge effectively, the system must be able to extract only the necessary knowledge from the knowledge base referring to the type and the domain of the problem to be solved. For this purpose knowledge must be well structured. All used knowledge is accumulated in the large knowledge base (Fig. 1).

2.2 Necessity of Knowledge Processing Language

The whole of the problem solving process is from accepting external representation of problems to generating solutions. In order to represent problems in the system a processing language is necessary. The language has to meet two conditions: it has to be usable for representing problems; and it has to be processable by computer processor. In ordinary computers only the procedural language is used both for processing by the processor and for representing problems. The knowledge base system, on the other hand, introduces the second language to separate the above two aspects, as well as a conversion mechanism between them. The second language is a declarative knowledge representation language. The conversion either in the declarative forms or from the declarative to the procedural form is necessary. This is the inference. It can be implemented as a procedural program on conventional computers.

The specification for the second language must be decided so that it can represent these conditions. It had to be suited for representing predicate including data structure as argument and also for describing higher-level operation such as knowledge for selecting object knowledge. KAUS (Knowledge Acquisition and Utilization System) has been developed for the purpose by our research & development team.

3 Adaptability of Learning Support System

To meet the condition of adaptability, it is necessity to represent the learner's understanding of learning domain. In this section, we discuss a learner modeling method that is applied to diagnostic techniques in artificial intelligence.

3.1 Issues of Learner Model

The performance of intelligent learning support system depends largely on how well it knows why the learner fails to solve problems. Because of the sophisticated interaction requires information about the learner, the system has to maintain some kind of model of the learner. This model may include cases about what has been done before or information about what the learner is believed to know. The process of gathering information about the learner is mostly referred to as cognitive diagnosis. Ohlsson has given a widely accepted definition of cognitive diagnosis: "cognitive diagnosis is the process of inferring a person's cognitive state from his or her performance" [4]. We consider that the point of learner model is to represent knowledge state of learner, especially his/her fails to solve problem. To satisfy this requirement, we focus diagnosis techniques.

A diagnosis is defined in terms of one or more reasoning steps that the learner cannot have solved problem. A major advantage of this approach is that it can be based solely on a model of these correct reasoning steps; no knowledge is required about the specific misconceptions that learners may have about the domain of learning. Instead we model all primitive inferences that are required to arrive at the correct solution. In addition, our approach to diagnosis of learner behavior exploits results from model-based diagnosis as it is defined in the field of artificial intelligence.

3.2 Model-based Learner Diagnosis with Case Base

Model-based diagnosis is a prominent area within artificial intelligence and emerged in the last about 15 years. The technique of model-based reasoning has been widely researched and accepted as the principal diagnosis in electronic circuit analysis, power station maintenance, medical diagnosis domains, etc. However, little emphasis has been put on its application to education or training system domain. The basic principle in model-based diagnosis is the description of system as a causal model. With the model at hand, the behavior predicated by the model is compared to the actually observed behavior. Since the predictions of the model are based on the assumption that the components work correctly, these assumptions may be partially dropped to accommodate for a detected behavior difference and thus diagnose faulty behavior.

However, there are some weaknesses in model-based diagnostic technique. The most serious weak point is the diagnosis time. It sometimes takes so much time to diagnosis. Therefore, we must be considering that it is necessary to model concerning the trade-off between the cost of a diagnosis time and its precision. Case-based reasoning, by contrast, excels in covering weak-theory domains, domains whose phenomena we do not yet understand well enough to record causality unambiguously. This feature allows case-based reasoning to be used in domains where model-based reasoning cannot be applied.

In the case-based reasoning, a reasoning engine remembers previous situations similar to the current one and uses them to help solve the new problem. However, case-based diagnostic technique has been criticized on many grounds. For example, that being specific to the system being diagnosed, they are non-constructive and that, having no analytic basis, the methods are restricted to specified faults and have a known level of competence. We think that the model-based diagnosis, being independent of the particular device descriptions, is intended to overcome these difficulties.

Therefore, we consider developing the approach of the model-based diagnosis system with case base. Model-based reasoning and case-based reasoning have the potential to complement each other quite well. However, no work has been done on specific issues of learner modeling using combine model-based reasoning with case-base. The outline of model-based learner diagnosis with case base is following. When the set of learner's behavior data input the diagnosis system, the diagnosis engine reasons the state of his/her knowledge consulting the diagnosis knowledge base include case base and object model base. The design of the model-based diagnosis system begins from describing the system as diagnosis object model. The system, which is a diagnosis object, is considered to be a set of domain models. The diagnosis object model that has knowledge of proper action, and the set of the behavior of learner as input value are given to a system. The first behavior of the system that received input is to seek whether there is a history about the same case in the case base. If the record to apply in the case is found, case base returns list of learner's knowledge, which should examine to diagnosis engine. Diagnosis engine does investigation about domain model of each record given to it, by comparing a simulation result in object model with the actual behavior of learner. Diagnosis process is finished if a trouble is recognized. When there was no record that complied with the input value in the case base, the process starts to use diagnosis domain object model. This domain object model has the hierarchical structure. A process begins from making the error model that one component in the extreme high class in the diagnosis object model is supposed to be out of order. The purpose of this process is to simulate using a made error model to examine whether the result of the simulation is the same as the behavior of learner. If there is no contradiction in the simulation result, the model-based reasoning is done again toward each domain knowledge model of the lower layer. In the same way, a diagnosis process is repeated until a trouble is recognized in knowledge component of the extreme lower layer. All process of diagnosis is knowledge processing by KAUS.

4 Discussion

The objective of this study was to develop a new intelligent learning support system, especially to focus two conditions; generality and adaptability. Authors in first propose the architecture of intelligent learning support system with large knowledge base to enough generality, which modeled by using multi-strata model. In second presented the model-based learner diagnosis to meet adaptability. All of the knowledge was represented by KAUS in intelligent learning support system that was assumed a part of the intelligent problem solving system. The issue of learner diagnosis is very important point to achieve adaptive instruction in intelligent learning support system. We proposed that fault diagnosis techniques be applied to infer the state of learner's knowledge. So we discussed the feature of diagnostic techniques, especially model-based reasoning with case base. Model-based reasoning appears to be a more promising technique than other knowledge-based methods because it can diagnose the faults that have not been pre-determined. Fails in learner's knowledge can be diagnosed automatically based on the models, which describe the correct behavior. However, because model-based approach reasons from the actual structure and function of knowledge, it is inefficient for some problems. Furthermore, obtaining domain models is sometimes either difficult or too complicated, whereas most of the fails can be diagnosed based on past experience, which is very effective if the rule base or the case base is either comparatively small or well-indexed. A better solution is a hybrid approach integrating some of the diagnostic approaches. A case base will be provided to access the solutions to some fails diagnoses occurred previously, of which the domain models are unavailable. For some diagnoses, their solutions and contexts can also be stored in the case base for reuse later. Frequently occurring fails can be diagnosed efficiently even by a few of heuristic diagnostic rules. We believe that such a hybrid diagnostic approach will perform better than any of them does. In order to achieve this goal; we have considered the division of object model and problem type. On this part, it is necessary to carry out examination that will be more profound in future work.

References

[1] Aida, T. & Ohsuga, S., "Intelligent Systems Must Be Able to Make Programs Automatically for Assuring the Practicality", Lecture Note in Artificial Intelligence 1415, pp.47-56, (1998).

[2] Costa, E, New Directions for Intelligent Tutoring Systems, Springer-Verlag Publishers, New York, (1991).

[3] Mandl, H. & Lesgold, A, (eds.), Learning Issues for Intelligent Tutoring Systems, Springer-Verlag Publishers, New York, (1988).

[4] Ohlsson, S., "Some principles of intelligent tutoring", Instructional Science, Vol. 14, pp.293-326, (1986).

[5] Ohsuga,S., "Multi-Strata Modeling to Automate Problem Solving Including Human Activity", Proc.Sixth European-Japanese Seminar on Information Modeling and Knowledge Bases, (1996).

[6] Sugaya,K. & Ohsuga,S., "Design of Intelligent Education System using Multi-Strata Modeling Concept", Proc. The Sixth International Conference on Computers in Education, Vol.1, pp.305-311, (1998).

[7] Sugaya,K. & Ohsuga,S., "Framework for Developing Intelligent Tutoring System Using Multi-Strata Model", Advanced Research in Computers and Communications in Education, Vo.1, IOS Press, pp.832-835, (1999).

[8] Wenger,E., Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge, Morgan Kaufmann Publishers, Los Altos, California, (1987).

Development of Intelligent Learning Support System with Large Knowledge Base

Katsuyuki SUGAYA and Setsuo OHSUGA
Department of Information and Computer Science, Waseda University
3-4-1 Ohkubo, Shinjuku-ku, Tokyo 169-8555, JAPAN
E-mail: {sugaya, ohsuga}@ohsuga.info.waseda.ac.jp

The objective of this paper is to present framework for developing intelligent learning support system with large knowledge base. Recently, the need for effective learning support and training is mounting, especially in industry or engineering fields, which demand the learning of complex tasks and expertise knowledge. Intelligent learning support system is being employed for this purpose, thus creating a need for cost-effective means of developing learning support systems. In this study, intelligent learning support system is assumed as a part of the intelligent knowledge management support system. The factors necessary for the intelligent learning support system discussed here are generality and adaptability. In order to achieve the goal, a new design of the system and learner modeling technique are discussed as well as a way of generating specific intelligent learning support system.

Keywords: **Intelligent System Design, Large Knowledge Base, Learner Model, Model-based Diagnosis, Knowledge Management**

1 Introduction

The purpose of this paper is to introduce a new framework for developing intelligent learning support system using large knowledge base. This system is a part of the intelligent systems that is developing to enable the expertise knowledge management.

In daily life, human has to interact with and reason about a large number of systems. This includes physical devices as well as non-physical systems. Also in professional work a growing number of people has to be trained in operating and designing large complex systems such as airplanes, nuclear power plants, and enterprises. Consequently, the goal of education or teaching may vary from inducing insight in the physical principles underlying the behavior of the device to teaching behavior analysis in the context of system design, operation, and maintenance. In addition, recently systems in the real world are becoming larger and more complicated. Rapid progress in science and technology has created a need for people who can solve complex problems and operate and maintain sophisticated equipment. In these situations, we, human beings, have to solve various types of problems using expertise in the large and complicated systems. Therefore the need for effective learning support or training is rising, given the increasing complexity of the workspace, especially in engineering or industrial fields.

Many computer assisted instruction techniques exist that can present instruction, and interact with students in a tutor-like fashion, individually, or in small groups [3]. The introduction of artificial intelligence technology and expert systems technology to computer assisted instruction systems gave rise to intelligent tutoring systems. In the intelligent tutoring system, for example, intelligent tutors that can model the learner's understanding of a topic and adapt the instruction accordingly [2]. Although intelligent tutoring systems research has been carried out for over 15 years, few tutoring systems have made the transition to the commercial market. Authors consider that some serious problems exist in the current methodology of developing intelligent tutoring systems. As an example, each system is developed independently, and tutoring expertise is hard-coded into individual systems. In particular, the problem of learner modeling technique exists as a basic issue. The system must have learner model that represents an estimate of the

BEST COPY AVAILABLE

learner current understanding of the domain knowledge to be used by tutor in order to give adaptive guidance and explanations to the learner. A number of learner modeling techniques have been developed [8]. However, not every model can be called complete expressing the learning condition of the learner. Hence, the motivation for this study comes from the need for effective intelligent tutoring systems, particularly development of more complete learner modeling technique.

For these problems like above we consider that the factors necessary for the intelligent learning support system discussed here are generality and adaptability. In order to achieve the goal, authors present a new framework of the intelligent learning support system those enough practical conditions. Several concepts are included in this study; expert knowledge management with large knowledge base, knowledge sharing, knowledge processing, model-based learner diagnosis, etc.

2 Expert Knowledge Management using Knowledge Base System

In this section, we introduce briefly the key concept of our knowledge base system. Our research groups have tried to solve various problems by knowledge-centered intelligent system. The main concept is Multi-strata modelling scheme [5]. This modelling scheme is applied many intelligent systems, and these systems rewarded with good results, e.g. automatic programming system [1]. And we considered that Multi-strata model is strongly support the development of intelligent tutoring systems [6][7].

2.1 Intelligent System with Large Knowledge Base

At first, we discuss to apply large knowledge base for the architecture of intelligent learning support systems, which can generate learning support systems for a wide range of domain.

In these days, with the developing of science and technology, the systems which human manages with are enlarged and more complicated. In particular, it is too difficult to transmit expert knowledge from expert engineer to novice engineers. In the engineering field, even a large system developed by many expert engineers. When the system grows lager and more complex, the knowledge that is needed to build the system is more specialized and subdivided. In these situations, some serious problems are occurred. For instance, it is difficult to communicate between expert engineer and another fields' engineers or novice one. In other words, it is too more expertise to transmission of expert knowledge from human to humans. For this reason, the expert knowledge hiding is occurred in some engineering companies.

When the knowledge is specified and subdivided, in the situation like classroom, it is not appropriate to transmit the knowledge from expert engineer to novice one e.g. next generation engineers. Therefore, we propose the transmission of expert knowledge through the large knowledge base system (Fig.1).

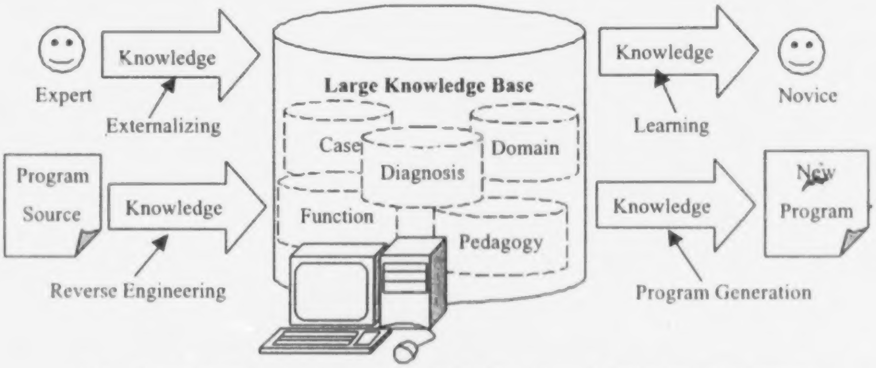


Fig.1: Knowledge Transmission using Intelligent Large Knowledge Base System

In this study, we consider that intelligent learning support system is a part of intelligent knowledge management support system. Moreover, we believe that knowledge management or learning support system is one of large and complex problem solving systems. The term problem is used here in a wide sense to mean what a person wishes to know or wants to do. There are various types of problems such as analysis, design control, decision-making, planning, and teaching. Most of them are not well dealt with by conventional software method but require the system a capability to find a solution itself in a large space. Since the space is open, self-controlled exploration in the space is necessary. The system must be provided with the various methods to solve the different type of problems, each of which is represented by a specific knowledge chunk. Furthermore, a complex problem concerns different problem domains and since a problem requires domain specific knowledge, the system must be provided with a global knowledge base including the various type of domain knowledge.

In order to use knowledge effectively, the system must be able to extract only the necessary knowledge from the knowledge base referring to the type and the domain of the problem to be solved. For this purpose knowledge must be well structured. All used knowledge is accumulated in the large knowledge base (Fig. 1).

2.2 Necessity of Knowledge Processing Language

The whole of the problem solving process is from accepting external representation of problems to generating solutions. In order to represent problems in the system a processing language is necessary. The language has to meet two conditions: it has to be usable for representing problems; and it has to be processable by computer processor. In ordinary computers only the procedural language is used both for processing by the processor and for representing problems. The knowledge base system, on the other hand, introduces the second language to separate the above two aspects, as well as a conversion mechanism between them. The second language is a declarative knowledge representation language. The conversion either in the declarative forms or from the declarative to the procedural form is necessary. This is the inference. It can be implemented as a procedural program on conventional computers.

The specification for the second language must be decided so that it can represent these conditions. It had to be suited for representing predicate including data structure as argument and also for describing higher-level operation such as knowledge for selecting object knowledge. KAUS (Knowledge Acquisition and Utilization System) has been developed for the purpose by our research & development team.

3 Adaptability of Learning Support System

To meet the condition of adaptability, it is necessity to represent the learner's understanding of learning domain. In this section, we discuss a learner modeling method that is applied to diagnostic techniques in artificial intelligence.

3.1 Issues of Learner Model

The performance of intelligent learning support system depends largely on how well it knows why the learner fails to solve problems. Because of the sophisticated interaction requires information about the learner, the system has to maintain some kind of model of the learner. This model may include cases about what has been done before or information about what the learner is believed to know. The process of gathering information about the learner is mostly referred to as cognitive diagnosis. Ohlsson has given a widely accepted definition of cognitive diagnosis: "cognitive diagnosis is the process of inferring a person's cognitive state from his or her performance" [4]. We consider that the point of learner model is to represent knowledge state of learner, especially his/her fails to solve problem. To satisfy this requirement, we focus diagnosis techniques.

A diagnosis is defined in terms of one or more reasoning steps that the learner cannot have solved problem. A major advantage of this approach is that it can be based solely on a model of these correct reasoning steps; no knowledge is required about the specific misconceptions that learners may have about the domain of learning. Instead we model all primitive inferences that are required to arrive at the correct solution. In addition, our approach to diagnosis of learner behavior exploits results from model-based diagnosis as it is defined in the field of artificial intelligence.

3.2 Model-based Learner Diagnosis with Case Base

Model-based diagnosis is a prominent area within artificial intelligence and emerged in the last about 15 years. The technique of model-based reasoning has been widely researched and accepted as the principal diagnosis in electronic circuit analysis, power station maintenance, medical diagnosis domains, etc. However, little emphasis has been put on its application to education or training system domain. The basic principle in model-based diagnosis is the description of system as a causal model. With the model at hand, the behavior predicated by the model is compared to the actually observed behavior. Since the predictions of the model are based on the assumption that the components work correctly, these assumptions may be partially dropped to accommodate for a detected behavior difference and thus diagnose faulty behavior.

However, there are some weaknesses in model-based diagnostic technique. The most serious weak point is the diagnosis time. It sometimes takes so much time to diagnosis. Therefore, we must be considering that it is necessary to model concerning the trade-off between the cost of a diagnosis time and its precision. Case-based reasoning, by contrast, excels in covering weak-theory domains, domains whose phenomena we do not yet understand well enough to record causality unambiguously. This feature allows case-based reasoning to be used in domains where model-based reasoning cannot be applied.

In the case-based reasoning, a reasoning engine remembers previous situations similar to the current one and uses them to help solve the new problem. However, case-based diagnostic technique has been criticized on many grounds. For example, that being specific to the system being diagnosed, they are non-constructive and that, having no analytic basis, the methods are restricted to specified faults and have a known level of competence. We think that the model-based diagnosis, being independent of the particular device descriptions, is intended to overcome these difficulties.

Therefore, we consider developing the approach of the model-based diagnosis system with case base. Model-based reasoning and case-based reasoning have the potential to complement each other quite well. However, no work has been done on specific issues of learner modeling using combine model-based reasoning with case-base. The outline of model-based learner diagnosis with case base is following. When the set of learner's behavior data input the diagnosis system, the diagnosis engine reasons the state of his/her knowledge consulting the diagnosis knowledge base include case base and object model base. The design of the model-based diagnosis system begins from describing the system as diagnosis object model. The system, which is a diagnosis object, is considered to be a set of domain models. The diagnosis object model that has knowledge of proper action, and the set of the behavior of learner as input value are given to a system. The first behavior of the system that received input is to seek whether there is a history about the same case in the case base. If the record to apply in the case is found, case base returns list of learner's knowledge, which should examine to diagnosis engine. Diagnosis engine does investigation about domain model of each record given to it, by comparing a simulation result in object model with the actual behavior of learner. Diagnosis process is finished if a trouble is recognized. When there was no record that complied with the input value in the case base, the process starts to use diagnosis domain object model. This domain object model has the hierarchical structure. A process begins from making the error model that one component in the extreme high class in the diagnosis object model is supposed to be out of order. The purpose of this process is to simulate using a made error model to examine whether the result of the simulation is the same as the behavior of learner. If there is no contradiction in the simulation result, the model-based reasoning is done again toward each domain knowledge model of the lower layer. In the same way, a diagnosis process is repeated until a trouble is recognized in knowledge component of the extreme lower layer. All process of diagnosis is knowledge processing by KAUS.

4 Discussion

The objective of this study was to develop a new intelligent learning support system, especially to focus two conditions; generality and adaptability. Authors in first propose the architecture of intelligent learning support system with large knowledge base to enough generality, which modeled by using multi-strata model. In second presented the model-based learner diagnosis to meet adaptability. All of the knowledge was represented by KAUS in intelligent learning support system that was assumed a part of the intelligent problem solving system. The issue of learner diagnosis is very important point to achieve adaptive instruction in intelligent learning support system. We proposed that fault diagnosis techniques be applied to infer the state of learner's knowledge. So we discussed the feature of diagnostic techniques, especially model-based reasoning with case base. Model-based reasoning appears to be a more promising technique than other knowledge-based methods because it can diagnose the faults that have not been pre-determined. Fails in learner's knowledge can be diagnosed automatically based on the models, which describe the correct behavior. However, because model-based approach reasons from the actual structure and function of knowledge, it is inefficient for some problems. Furthermore, obtaining domain models is sometimes either difficult or too complicated, whereas most of the fails can be diagnosed based on past experience, which is very effective if the rule base or the case base is either comparatively small or well-indexed. A better solution is a hybrid approach integrating some of the diagnostic approaches. A case base will be provided to access the solutions to some fails diagnoses occurred previously, of which the domain models are unavailable. For some diagnoses, their solutions and contexts can also be stored in the case base for reuse later. Frequently occurring fails can be diagnosed efficiently even by a few of heuristic diagnostic rules. We believe that such a hybrid diagnostic approach will perform better than any of them does. In order to achieve this goal; we have considered the division of object model and problem type. On this part, it is necessary to carry out examination that will be more profound in future work.

References

[1] Aida, T. & Ohsuga, S., "Intelligent Systems Must Be Able to Make Programs Automatically for Assuring the Practicality", Lecture Note in Artificial Intelligence 1415, pp.47-56, (1998).

[2] Costa, E, New Directions for Intelligent Tutoring Systems, Springer-Verlag Publishers, New York, (1991).

[3] Mandl, H. & Lesgold, A., (eds.), Learning Issues for Intelligent Tutoring Systems, Springer-Verlag Publishers, New York, (1988).

[4] Ohlsson, S., "Some principles of intelligent tutoring", Instructional Science, Vol. 14, pp.293-326, (1986).

[5] Ohsuga,S., "Multi-Strata Modeling to Automate Problem Solving Including Human Activity", Proc.Sixth European-Japanese Seminar on Information Modeling and Knowledge Bases, (1996).

[6] Sugaya,K. & Ohsuga,S., "Design of Intelligent Education System using Multi-Strata Modeling Concept", Proc. The Sixth International Conference on Computers in Education, Vol.1, pp.305-311, (1998).

[7] Sugaya,K. & Ohsuga,S., "Framework for Developing Intelligent Tutoring System Using Multi-Strata Model", Advanced Research in Computers and Communications in Education, Vo.1, IOS Press, pp.832-835, (1999).

[8] Wenger,E., Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge, Morgan Kaufmann Publishers, Los Altos, California, (1987).

Development of the Web-based classroom system to be implemented by the teachers

Go Ota* and Kanji Akahori*
**Cooperative Research Center Saitama University*
***Tokyo Institute of Technology*
*E-mail: *gohome@jona.or.jp **akahori@cradle.titech.ac.jp*

The Japanese ministry of Education made an announcement that a new curriculum "Information and Computer" will be introduced nation-wide in Japan from the year 2002. Accordingly, all the schools have been rushing to deploy the personal computers and prepared to connect to the Internet through 2001. While the scope of this project aims at covering 40000 or more schools, there exists the two major problems: 1) The number of teachers who have expertise in handle the PC and the Internet, are too far short in proportion to the number required. 2) Dial-up networking prevents the students from having access to the Internet any time when they want. With a view to overcoming these problems, we have designed and developed the Intranet system or "micro Internet for classroom: mIc". The "mIc" is developed and designed to incorporate the various functions such as web-mail, electronic bulletin board "BBS", mailing list, search engine, web video conference and etc. Since "mls" consist of Microsoft Active Server Pages (ASP), it can be used from Web browsers and custom-tailored at ease.

Keywords: **Intranet, Collaboration, Video-conference, BBS**

1 Introduction

The Japanese ministry of Education made an announcement that a new curriculum "Information and Computer" will be introduced nation-wide in Japan at both the elementary school and the junior high school in 2003 and at the high school in 2002 respectively. Accordingly, all the schools have been rushing to deploy the personal computers and are prepared to connect to the Internet through 2001. While the PC have been gradually and extensively, it seems quite obvious that far small number of the teachers can handle the PC and the Internet to the contrary.

The Minister of Education has been sending the computer engineers or other computer technical personnel to school since 1994 with a view to training the teachers about the computer and the Internet. They are also required to see to it that both the teachers and the students can implement the PC and the Internet smoothly without any problems. Additionally, The Ministry has been initiating their own training programs for the teachers as well. While the project is supposed to cover 40000 schools or more, it has been experiencing the extreme difficulties of the shortage in the engineers and the technical staffs to reach out all the teachers in 4000 schools or more. It has been experiencing the difficulties as well as that Dial-up networking prevents the students from having access to the Internet any time when they want.

Despite these difficulties, it seems quite viable that all the students will get accustomed to the computer and the Internet at the earliest convenience. We, therefore, have designed and developed the Intranet System(micro Internet for classroom: mIc)

2 Design of mIc

This system "mIc" is developed and designed for both the teachers with least knowledge about the PC and the Internet, and the students as well to learn the various functions.

(a) Simulation of the Internet.
We are of an opinion that the E-mail and Electronic Bulleting Board shall be viable tools for "collaboration" among the students. Should the students require any information from the Internet, the search engine shall be inevitable to learn as well. We, therefore, have designed to incorporate these functions in the system. The teachers simply use the system without any other programs and the students can experience those functions as if they were connected to the Internet.

(b) Web-based easy operation.
The teachers can use "mlc" from Web browser. Therefore, should the teachers use the system, they can create new BBS, mailing list and registration of the students on Web based. As far as the teachers will use solely "mlc", the profound knowledge about the Internet server and the program of CGI is not necessary.

(c) Customization.
The curriculum of "Information and Computers" varies depending on the computers deployed ,the network system applied, and the objective of the education for PC & the internet in each school respectively. The system "mlc" can be customized by merely changing the text-files.

3 Structure of mlc

Considering the Standardizing the server of the average school environment, "mlc" will be installed in WindowsNT server or Window98. Please take note that less than 10 people can work with Window98 simultaneously.

3.1 ASP and COM

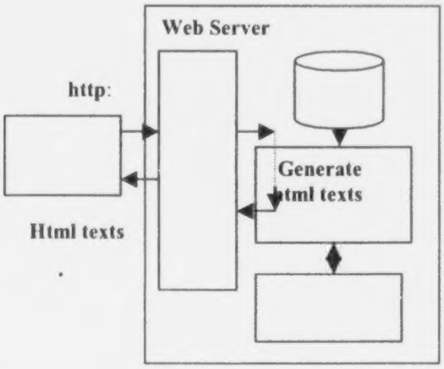


Figure 1: The Action of ASP

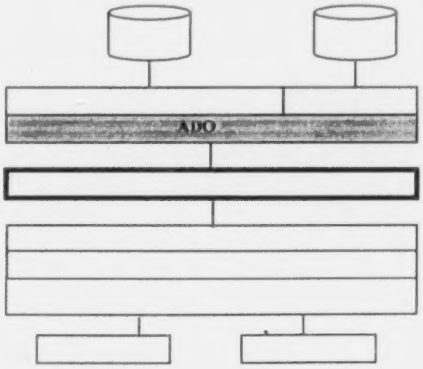


Figure 2: The structure of mlc

The system "mlc" consists of Microsoft Active Server Pages(ASP) which is the server-side execution environment. The ASP can run scripts and Component Object Model(COM) on the server . It can also easily create the dynamic contents and the powerful Web-based applications. The COM is the Microsoft software architecture that allows application to be built from binary software components. Windows itself and many other applications such as WORD, EXCEL and etc. are consisted of the COM.

Figure 1 shows the process of "mlc". ASP files appears to be the same as the HTML files but it includes additionally VBscripts or Javascripts, which call COM. At first, a browser makes a request to the server to send an ASP file in such a manner as to the HTML file. Secondly, the server executes ASP file and Bvscripts or Javascripts At last, the server send these to a browser. By using ASP, a browser only interprets common HTML without executing scripts in the client environment. Figure 2 shows the structure of "mlc". We have applied to some COM, which have access to a database, a browser, files, and a mail server. ADO is the database access COM and the system uses Microsoft Access or SQL Server.

3.2 Setup of mlc

The system "mlc" can be easily installed by simply copying the ASP files in such a manner as for HTLM files .The teacher will be required to edit the "mlc"' configuration file which contains such information as URL, the install path and etc. Should a teacher wish to display some comments enabling the students to take note for their reference, he simply input the comments in the text-file corresponding to the exact page. The "mlc" can build more than one system in one server by creating more than one data base file.

4 System function

The functions of "mlc" will be detailed as follows;

4.1 Registration

The teachers can register the students with the use of browser. They can register even many number of students at once with the use of EXCEL or ACCESS. If the teachers will use BBS and E-mail via other programs than "mlc", they will be required to register newly each time they change the application.

4.2 System Menu

Three different user modes are available in the menu , one for a teacher , one for students and one for a guest respectively. The teacher can customize the menu for each mode. Should the teacher not use the mailing list, he can simply edit the configuration file to turn off the flag of the mailing list and the menu eventually will not display the button of the mailing list.

4.3 Web mail

The system "mlc" has two different Web mail modes whose user interface are the same , the one simulation mode and the other SMTP/POP3 mode. While the simulation mode will not actually allow to send or receive mails via the Internet, it will allow to simulate the mail functions without the mail server. Should you have the mail server and use the SMTP/POP3 mode, it will allow to send or receive mails via the Internet as the regular web mail.

4.4 Electronic bulletin board (BBS)

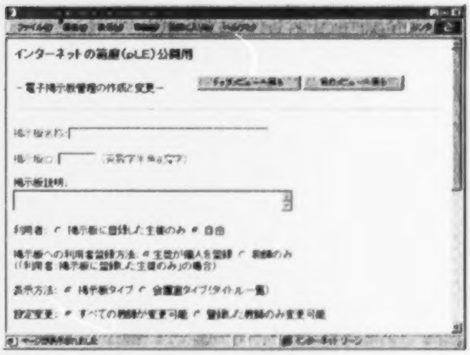


Figure 3: Screen image of Defining BBS

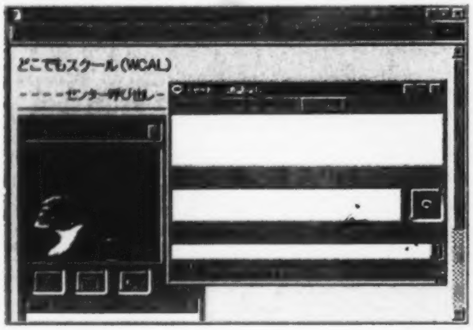


Figure 4: Screen image of Video conference

The system "mlc" allows to set up more than one bulletin board. Should the teacher wish to create a new BBS, he will be required to simply define the BBS on the browser and no new program will be necessary(Figure 3). "mlc" allows to set up the users' list covering the users who can have the access only in the BBS. The users' list can be selected in accordance with the student attribution such as Class , Group and etc.

4.5 Mailing list

The operation of the mailing list will follow the same manners as mentioned above for BBS.

4.6 Search engine

Since "mlc" has a directory service like "YAHOO" , the teachers and the students can add any new URL to the directory for their reference. If "mlc" is installed in WindowsNT server with Microsoft Index Server, the text-matching search engine can be used. The attention is drawn that "build-up of HP" has become one of the most important curriculum in Japan . The student can register their own HP's in the directory of "mls" and can subsequently search them in the classroom.

4.7 Web Video conference

Since the Video conference is very efficient and effective tool in term of the international communication, we have designed to incorporate the function "Web Video conference" in the system so as to suffice in this respect(Figure 4). A student can communicate with other students and visualize them via web video conference and refer to the data interactively via web data conference. Data conference allow the students to collaborate on "chat", "whiteboard" and "program sharing" without Video and Audio. Since the web videoconference is based on Microsoft Netmeeting 3.0 Active X, the multipoint data conference is possible and thus more than one student can participate the meeting simultaneously.

4.8 Generator of the questionnaire

Understanding strongly the importance of the questionnaire so as collect of the opinion from the students for various topics, "mlc" is designed to generate automatically the questionnaire in the form of HTML and ASP files. The teacher can easily make these files by filling in to the points raised as question on the web pages . The form filled in by the students can be saved to the text file in the form of the spreadsheet such as Excel.

5 Further development(future work)

We have already started to introduce the system"mlc" at schools ranging from the junior high school through the university . Having learnt from the experience, it seems very obvious that the teachers can make BBS and use search engines at ease. Through the continued experiments, we are prepared to improve the system further.

mlc Web Site (In Japanese)
URL www.jona.or.jp/~gohome

References

[1] Michael,G., Greg Merley, Distance Education A System View, Wadsworth Publishing Company, (1996)
[3] Go,O.,Kanji ,"Development of integrated Collaboration System with BBS and mailing list"(1999)
[2] Go,O.,Kanji, A.,"Development of School intranet system baeed on COM and ASP", (1999)

Implementation of An Object-Oriented Learning Environment Based on XML

Geng-Hong Su, S. S. Tseng*, Chang-Jiun Tsai, and Jing-Ru Cheng
Department of Computer and Information Science
National Chiao Tung University
Hsinchu 300, Taiwan, R.O.C.
Tel: +886-3-5712121 ext. 56658
Fax: +886-3-5721490
E-mail: sstseng@cis.nctu.edu.tw

In [12], we proposed the architecture of Object-Oriented Course Construction System (OOCCS). Based upon OOCCS, dynamic and individualized course frameworks can be constructed from the teaching objects in accordance with the aptitudes of students. Besides, the teachers can change the content of teaching materials easily by just inserting, deleting or modifying the related teaching objects. In this paper, we design an authoring system to assist teachers in construct teaching objects by using XML technology, and construct some mathematical teaching objects for showing the dynamic and individualized course frameworks.

Keywords: **Object-Orientation, Web-based Learning Environment, XML**

1. Introduction

The web-based educational systems are becoming more and more popular over the world. Several approaches, which are used to organize the teaching materials appropriately, have been developed in the past ten years [1][2][4][6][8][11]. One of them is just putting all of teaching materials on the web looked just like an electronic book although some friendly user interface or interesting multimedia has been added [1][2][8][11]. [4] and [6] provided the evaluation mechanismwhich works at the end of each section of course materials to find out what instructional objectives the students do not learned well. According to these evaluation results, the system can offer the remedial teaching materials properly for students to learn again. However, in above two approaches, the students always need to learn all teaching materials at least once no matter how the teaching materials are suitable for them or not when they enter a new section. Therefore, we are interested in developing a tutoring system, which can offer different teaching materials for different students in accordance with their aptitudes. As shown in Figure 1, the traditional course model usually arranges the teaching materials in sequential and monotonous way. It means that the individualized course framework may not be offered in accordance with their aptitudes.

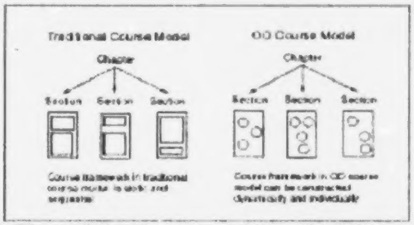

Traditional Course Model
Chapter
Section
Section
Section
Course framework in traditional course model is static and sequential
OO Course Model
Chapter
Section
Section
Section
Course framework in OO course model can be constructed dynamically and individually

Figure 1. The traditional course model vs. the OO course model

---

* Corresponding author.

BEST COPY AVAILABLE

In last year, we proposed the architecture of Object-Oriented Course Construction System (OOCCS) [12]. Based upon OOCCS, dynamic and individualized course frameworks can be constructed from the teaching objects in accordance with the aptitude of students. Besides, the teachers can change the content of teaching materials easily by just inserting, deleting or modifying the related teaching objects.

In this paper, we design an authoring system to assist teachers in construct teaching objects by using XML technology, and construct some mathematical teaching objects for showing the dynamic and individualized course frameworks.

2. Related Work

In the traditional course model, the arrangement of the teaching materials in a section is sequential and monotonous. In this way, without appropriate segmentation and labeling on teaching materials, it is difficult for an individualized tutoring system to offer appropriate teaching materials for students in accordance with their aptitude. The "segmented materials & materials attributes" relationship actually is very similar to the "class & class members" relationship in object-oriented mechanism. Therefore, object-oriented course model [12] have been proposed to model the segmented teaching materials. In this model, the teaching materials of a section are divided into several segments according to the instructional objectives defined by educational experts. Besides, each of them is attached with four labels, the background knowledge, instructional objective, the learning level and the difficulty level, which are described detailedly in Table 1.

Table 1. Instructional Objective Class Definitions

CLASS MEMBER	DEFINITION	INITIALIZE
Background Knowledge	■ The prerequisite instructional objectives before learning the background knowledge.	Base class
Instructional Objective	■ The gained knowledge after learning.	Base class
Learning Level	■The appropriate level of instructional objective for students. ■ Smaller learning level is more basic for students.	Base class
Difficulty Level	■ The difficulty of a specific object.	Sub-class

As shown in Fig. 2, the object-oriented course model basically is a two-tier architecture: the fist tier is the instructional objective base-class and the second tier is the teaching material sub-class. Based upon the inheritance property of object-oriented concept, the attribute of teaching material sub-class can be easily inherited from the corresponding base class. When teachers want to transform a teaching material into a teaching object, they must select an appropriate instructional objective base-class and determine the instructional objective and the difficulty level. Then the system would automatically derive a teaching material sub-class and generate a concrete teaching object from the sub-class. This transformation process is repeated until all the original teaching materials are transformed into teaching objects.

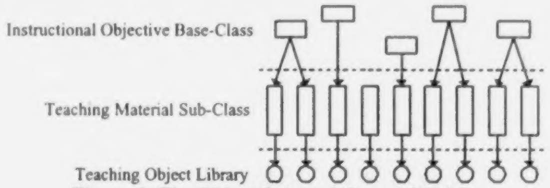

Instructional Objective Base-Class
Teaching Material Sub-Class
Teaching Object Library

Figure 2. The Object-Oriented Course Model

In the first tier, for a given section of a teaching material, an instructional objective base-class with the value of class members can be constructed according to the instructional objective defined by educational experts. In the second tier, for a teaching material sub-class, the class members are initialized from the base class, such as instructional objective, background knowledge and learning level, are completely inherited from the base classes through the inheritance paradigm The class member, difficulty level, is initialized in the teaching material sub-class according to the content of a particular teaching material. The instructional objective base-class in the first tier and the teaching material sub-class in the second tier compose the course template library in the object-oriented course construction system.

Based upon object-oriented course model, the architecture of object-oriented course construction system

(OOCCS) has been proposed in last year. As shown in Fig. 3, there are five components, the Course Template Library, the Teaching Object Library, the Learning Record Database, the Individualized Course Construction Engine, and the Evaluation Center, in OOCCS. Furthermore, the individualized course construction algorithm and the course framework revision algorithm has been proposed too.

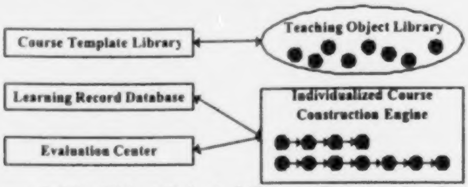


Figure 3. The Object-Oriented Course Construction Systems

3. Data Representation

In this paper, the Extensible Markup Language (XML) [3] is used as the tool of data representations for the content of teaching materials. Based on XML document model, the Document Type Definition (DTD) and schema can be used to evaluate whether the teaching materials are well formed and valid. Besides, the Extensible StyleSheet Language (XSL) [5] can be used to translate one XML document into several different representations. To compose the XML-format teaching materials with different kinds of XSL-format layout templates, different course frameworks can be generated according to students' aptitudes from only one set of teaching materials. Most important of all, under the XML mechanism, the teaching materials own the following benefits including Extensibility, Machine-Readable, Validation, and Convenience.

3.1 The DTD and Schema

Schema in the database is used to determine the order and data type of each field. Similarly, a Document Type Definition (DTD) or an extensible set of rules, also called a schema, is used to validate an XML document by the XML parser. An XML document that follows a DTD or a schema is said to be valid. The Document Type Definition (DTD) describes the structure of an XML document. The DTD includes the information about what elements must be present, which ones are optional, what their attributes are, and how they can be structured with relations to each other. In object oriented course model, four attributes Major Concept, Background Knowledge, Difficulty Level, and Learning Level are included in a teaching object. The corresponding DTD of the teaching object is as follows.

```
<?XML version="1.0">
<!DOCTYPE TeachingMaterial [
    <!ELEMENT TeachingMaterial        (To+)>
    <!ELEMENT To                      (Attributes)>
    <!ELEMENT Attributes              (MajorConcept,BackGroundKnowledge,DifficultyLevel,LearningLevel)>
    <!ELEMENT ContentType             (#PCDATA) >
    <!ELEMENT LinkPath                (#PCDATA) >
    <!ELEMENT MajorConcept            (#PCDATA) >
    <!ELEMENT BackGroundKnowledge     (#PCDATA) >
    <!ELEMENT DifficultyLevel         (#PCDATA) >
    <!ELEMENT LearningLevel           (#PCDATA) >
]>
```

The schema for the XML document is used to define the syntactic structure and partial semantics of XML document types. As such, the schema is an alternative to XML DTDs and can be used to define the same class of document types (with the exception of external parsed entities). Besides, the schema extends the language of DTDs by the following supporting.

- **An extensive (and extensible) set of data types.**
- **Inheritance among element types.**
- **Namespaces.**
- **Features to enable robust distributed schema management.**

The XML Schema of a teaching object is designed in Fig. 4. In the schema for teaching object, we define an element "Attributes" with content type "Attribute" in the "Teaching Material" element. For an element "Attributes", four sub-elements Major Concept, Background Knowledge, Difficulty Level, and Learning Level are included.

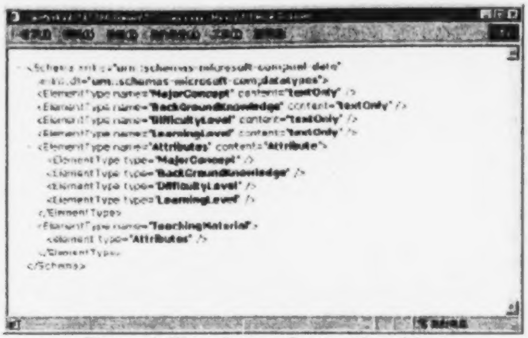


Figure 4. The schema of a teaching object

3.2 The Extensible StyleSheet Language (XSL)

The XML document is a tree-like structure containing the elements, attributes, entities, and so on. The XSL documents takes the tree generated by XML, called the "source tree", and create a new "result tree" that includes all of the objects to be output along with proper formatting information. Through the Document Object Model (DOM) API, an XSL processor can be invoked to apply formatting information to these objects and output them in the browser.

Style-sheet developer usually considers two parts, pattern and template. The pattern is used to match with elements and attributes and the template is used to generate the result tree. For example, the template rule as shown in following looks for "TeachingMaterial" elements in the source tree and are placed inside an <xsl:stylesheet> element. When the processor finds such element, the formatting portion of the template rule is applied to the paragraph content.

```
<xsl:stylesheet xmlns:xsl = "http://www.w3.org/TR/WD-xsl" xmlns:fo=http://www.w3.org/TR/WD-xsl result-ns="fo ">
    <xsl:template match="TeachingMaterial">
        <fo: black font-size = "10pt" space-before="12pt">
            <xsl:apply-templates/>
        </fo:block>
    </xsl:template>
</xsl:stylesheet>
```

4. Experiments

As shown in Fig. 5, the architecture including the Teacher Module, Logic Module, and Student Module is designed for the object-oriented learning environment. Besides, in these three modules, there are four components including the Authoring Interface, the Teaching Material Importing Interface, the Course Construction Engine, and the Course Formatting Interface.

- **The Teaching Module**
  An Authoring Interface is designed for teachers to mark-up their teaching contents with corresponding tags.
- **The Logic Module**
  The teaching material importing interface is designed to accept, parse, and finally insert XML-format teaching contents into teaching material database. Besides, a course construction engine is also designed to organize the different course frameworks according to students' aptitudes.
- **The Student Module**
  The course-formatting interface is designed for students to generate course framework easily according to their learning situations.

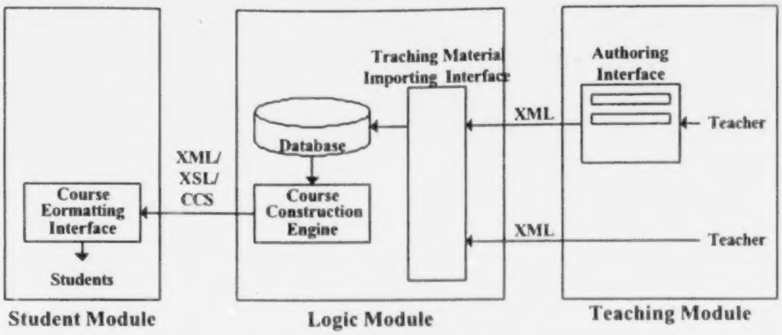

Traching Material Importing Interface
Authoring Interface
XML
Teacher
Database
XML/ XSL/ CCS
Course Eormatting Interface
Course Construction Engine
XML
Teacher
Students
Student Module
Logic Module
Teaching Module

Figure 5. The object oriented learning environment

4.1 The Authoring System

For the teaching objects in the object-oriented course model, they contain not only the teaching contents but also the attributes including the background knowledge, the instructional objective, learning level, and the difficulty level. According to these attributes, it implies these attributes are very important because the course construction engine can organize different course framework for different students. A friendly authoring interface is designed as shown in Fig. 6 to help teachers to author and combine these attributes with teaching contents smoothly. Besides, for the consideration of extensibility, machine-readability, and validation, we choose the Extensible Markup Language (XML) to markup the teaching contents with important meta-information (attributes).

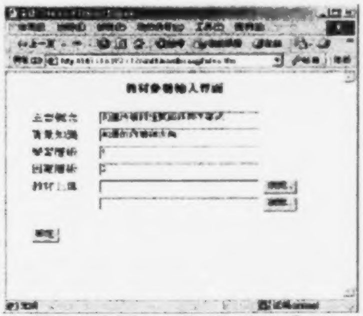


Figure 6. The authoring interface

The teachers need to fill-in the attributes of the uploaded teaching contents and the XML-formatted teaching materials would be automatically created. According to these attributes shown above, the following XML document in Fig. 7 would be generated. Besides, XML schema would be used to check whether the generated XML document is well formed and valid. This checking process is very important for making sure the structure of inputting data is right. Fig. 8 shows the XML schema for the teaching materials.

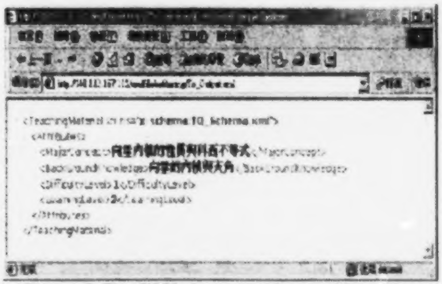


Figure 7. The XML document for teaching object

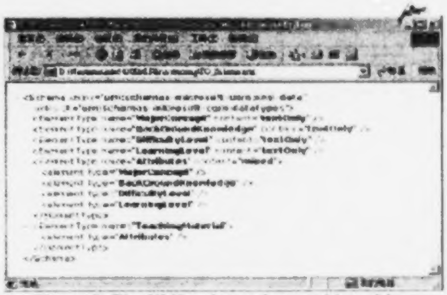


Figure 8. The XML schema for teaching object

4.2 The Teaching Material Importing Interface

The teaching material importing interface is designed for importing and parsing the XML-format teaching materials first and then inserting them into teaching material database.

■ Parse the teaching materials

For parsing the teaching materials, the Microsoft's XML parser, which resides in a file called Msxml.dll and is a Component Object Model (COM) object, could be used in this processing with any ActiveX-compliant environment, including VB and Active Server Page. The following shows the codes of our experiment using the Active Server Page with the Microsoft ' s parser.

```
Set objXML = Server.CreateObject("Microsoft.XMLDOM")
  If Not objXML.Load(Server.MapPath("TO.xml")) Then
   With objXML.parseError
        If .errorCode <> 0 Then
                Response.write "Parse Error"
        Else
                Response.write "Error Occured.."
        End If
        End With
   End If
   Set m_Root=objXML.documentElement
   Set KNode=m_Root.selectNodes("Attributes")
   Response.write "· · · · · · · · · · · · · · · · · · <br><br>"
   For Each node in KNode
        Response.write   node.selectSingleNode("MajorConcept").Text      Response.write
        node.selectSingleNode("BackGroundKnowledge").Text
        Response.write node.selectSingleNode("LearningLevel").Text      Response.write
        node.selectSingleNode("DifficultyLevel").Text
   Next
```

In the parsing process, an objXML object is first created as Microsoft DOM document and then uploaded teaching materials would be loaded. By checking the "parseError" property to objXML, the uploaded teaching material would be checked whether is well formed and valid. If the checking result is right, several methods including the "DocumentElements", "SelectNode" and the "SelectSingleNode" in the DOM objects can be called to parse the XML-format teaching materials and retrieve the attributes.

■ Insert the parsed result into database

After parsing the uploaded teaching materials, system would insert the teaching materials and the meta-information (attributes) into database. Because database allows difficult data formations to be created dynamically from the same data, the XML-format teaching materials are imported into the database for manipulating teaching materials efficiently.

After parsing XML-format teaching materials and inserting the parsed result into database, the feedback is shown in Fig. 9. The importing interface can parse the uploaded teaching materials and check whether the teaching materials are well formed and valid. Therefore, teachers can edit the teaching materials in the client side and then upload the final teaching materials through the importing interface into server. By this way, the distributed authoring environment is obtained.

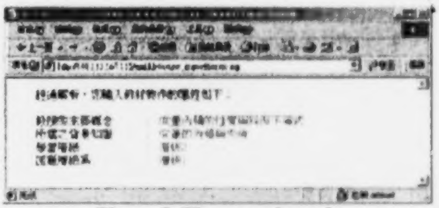


Figure 9. The parsed result

BEST COPY AVAILABLE

4.3 The Course Construction Engine

The Course Construction Engine is designed to organize the teaching objects according to the Individualized Course Construction Algorithm and the Course Framework Revision Algorithm [12]. By applying these algorithms, the individualized course framework can be constructed from the teaching objects stored in the database and can be organized in accordance with the aptitudes of the students.

In the construction process, the generated course frameworks are saved in the XML format. This means that the course framework contains not only the teaching contents but also the attributes (meta-information) including the Background Knowledge, the Instructional Objective, Learning Level, and the Difficulty Level. These attributes are encoded in the defined XML tags and can assist the course-formatting interface in changing the representation for the course frameworks dynamically and easily.

4.4 The Course-Formatting Interface

The course-formatting interface is designed for presenting the course frameworks, which are organized by the course construction engine. In the interface, several different XSL-format layout templates would be designed. Students can choose the desired layout template, and this template can be used to translate the XML-format course framework into readable teaching materials for reading and learning. The following results show the same course framework with Learning Level 1 and Learning Level 2 in Fig. 10 and Fig. 11, respectively.

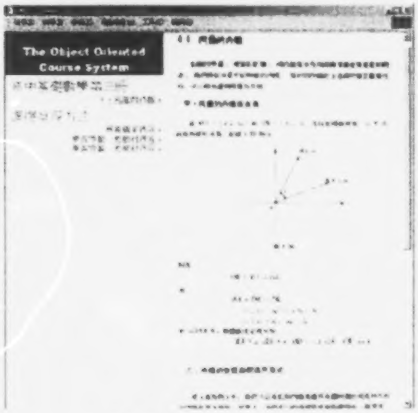


Figure 10. Learning Level 1

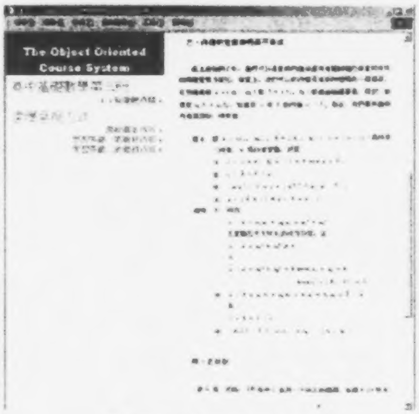


Figure 11. Learning Level 2

As shown in the above results, through the layout templates in the course-formatting interface, an XML-format course framework can be constructed into different layouts automatically. It is important for object oriented course model when students want to choose other layout of the course framework, because the different course frameworks can be provided by changing the layout templates easily.

5. Concluding Remarks

In this paper, based upon object oriented course model, we design an authoring system to assist teachers in construct teaching objects by using XML technology. All of teaching objects would be stored in database for further usage of OOCCS. OOCCS can offer different course frameworks to the students according to their aptitudes. When the students do not learn some major concepts well, OOCCS can precisely offer the related teaching objects for them to learn again.

In the experiment, the authoring system, the teaching material importing interface, the course construction engine, and the course-formatting interface are implemented under object-oriented and XML concepts. Teachers can edit the teaching materials in the client and then upload into server easily. After the course construction engine and course formatting interface transform the teaching materials into teaching objects, OOCCS can organize course framework in accordance with students' aptitudes automatically.

However, there are still some disadvantages including the "additional teaching material preparation time" and the "not suitable for all curriculums". In the future, we will try to develop a friendly front-end on the WWW to

BEST COPY AVAILABLE

minimize the effort needed to prepare teaching materials, and then apply object-oriented course model to the mathematics curriculum to examine how they work.

Acknowledgment

This work was supported by the National Science Council of the Republic of China under Grant No. NSC88-2213-E-009-078.

References

[1] Alessi, S. M. and Trollip, S. R., "Computer-based instruction: Methods and development (2^{nd} ed.)," Englewood Cliffs, NJ: Prentice-Hall, (1991).

[2] Antal, P, "Animated explanations using adaptive student models," Proc. of INES '97, Page(s): 573 –576, (1997).

[3] Bray, T., Paoli, J., and Sperberg-McQueen, C. M., "Extensible Markup Language (XML) 1.0," World Wide Web Consortium (W3C), http://www.w3.org/TR/1998/REC-xml-19980210 (Feb 1998).

[4] Chou, C., "A computer logging method for collecting use-reported inputs during formative evaluation of computer network-assisted distance learning," Proc. of ED-Media' 96 Conference, (1996).

[5] Clark, J., and Deach, S., "Extensible StyleSheet Language (XSL) Version 1.0," World Wide Web Consortium (W3C), http://www.w3.org/TR/1998/WD-xsl-19981216 (Dec 1998).

[6] Campbell, J.O., "Evaluating costs and benefits of distributed learning," Proc. of 27th Annual Conference, (1997).

[7] Essenius, R. P. and Wissenburgh, C., "Adaptive computer-based training in electronic engineering," Proc. of MCMC '97, Page(s): 73 –82, (1997).

[8] Hwang, G.J., "A Tutoring Strategy Supporting System for Distance Learning on Computer Networks," IEEE Transactions on Education, Volume: 41, No. 4, Page(s): 343 –343, (1998).

[9] Principe, J.C., Euliano, N., Lefebvre, C., "An interactive learning environment for adaptive systems instruction", Proc. of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, Volume: 3, Page(s): 1901 -1904, (1998).

[10] Srisethanil, C. and Baker, N., "ITS-Engineering: providing adaptive teaching in the engineering tutor." Proc. of Frontiers in Education Conference, Volume: 1, Page(s): 2a3.22 -2a3.26.

[11] Sun, C. T. and Chou, C., "Experiencing CORAL: design and implementation of distance cooperative learning," IEEE Transactions on Education, Volume: 39, No. 3, Page(s): 357 –366, (1996).

[12] Su, G. H., Tseng, S. S., Tsai, C. J., and Zheng, J. R., "Building an object-oriented and individualized learning Environment on the WWW," Proc. of ICCE' 99, Page(s) 728-735, Chiba, Japan, (1999).

[13] Vassileva, J., "Dynamic Course Generation on the WWW," Proc. of AI-ED' 97, IOS: 498-505, (1997).

[14] Weiss, D.J. and Kingsburg, G.G., "Application of Computerized Adaptive Testing to Educational Problems," Journal of Educational Measurement, Volume: 21, Page(s) 361-375, (1984).

[15] Yu, P.T. and Shyi C.W, "Intelligent Computer-Assisted Learning System: A Mental-Model Approach", project report of National Science of ROC, NSC 85-2511-S-194-005, (1999).

A Web-based EFL Writing Environment: Integrating Information for Learners, Teachers, and Researchers

David Wible*, Chin-Hwa Kuo, Anne Liu* and Nai-lung Tsao****
** Graduate Institute of Western Languages and Literature*
Tamkang University
Tamsui, Taipei Hsien, Taiwan
dwible@mail.tku.edut.w
lieor@ms7.url.com.tw
***Computer and Networks (CAN) Laboratory*
Department of Computer Science and Information Engineering
Tamkang University
Tamsui, Taipei Hsien, Taiwan
chkuo@mail.tku.edu.tw
g6190173@tkgis.tku.edu.tw
Taiwam

With the rise in the popularity of web-based education, there is a pressing need for the design of web-based systems that are domain-specific. This need is particularly acute for the domain of second language education, where generic web-based systems fall short of fulfilling the potential of the Internet for meeting the particular challenges faced by language learners and teachers. A novel interactive online environment is described which integrates the potential of computers, Internet, and linguistic analysis to address the highly specific needs of second language composition classes. The system accommodates learners, teachers, and researchers. A crucial consequence of the interactive nature of this system is that users actually create information through their use, and this information enables the system to improve with use. Specifically, the essays written by users and the comments given by teachers are archived in a searchable online database. Learners can do pinpoint searches of this data to understand their individual persistent difficulties. Teachers can do the same in order to discover these difficulties for individual learners and for a class as a whole. The modular system provides interfaces with functions to facilitate an array of user tasks such as teachers' correction of essays and learners' writing and revision processes. Researchers' error analysis of learner essays feeds an active online help function as well.

Keywords: **applications in subject areas; architectures for educational technology system; interactive learning environments**

1 Introduction

The purpose of this paper is to describe one module in a highly integrated language learning environment called IWiLL (Intelligent Web-based Interactive Language Learning: http://www.can.tku.edu.tw/iwill) The module within IWiLL which we focus on in this paper is a novel web-based writing environment designed for EFL composition classes.

The design of the IWiLL writing environment is based upon certain general assumptions. First, the ideal online system for EFL writing classes should be interactive. Second, the system should exploit the computers' capacity to track the content of the interactions between users and to enable users to do pinpoint searches of the record of these interactions. This feature makes available invaluable information that can serve as a cumulative source of insight for both learners and teachers, information which in traditional writing classrooms remains scattered, ephemeral, and highly inaccessible. Third, while the system described here is designed for second language writing classes, it is more accurately seen as one component in an integrated language learning environment that includes other skills, such as reading and listening (Kuo, Wible, and Chio, 2000; Kuo, et. al., 2000). The modularized and integrated design is intended to accommodate recent trends in language pedagogy which view language skills as best learned in an integrated whole rather than as a set of separate, independent skills. Finally, while the IWiLL environment is designed specifically to meet the needs of certain type of language course (second language composition) it is intended to provide as much freedom as possible for teachers within this domain to use their own approaches and materials of their choice.

2 The Organization of the System

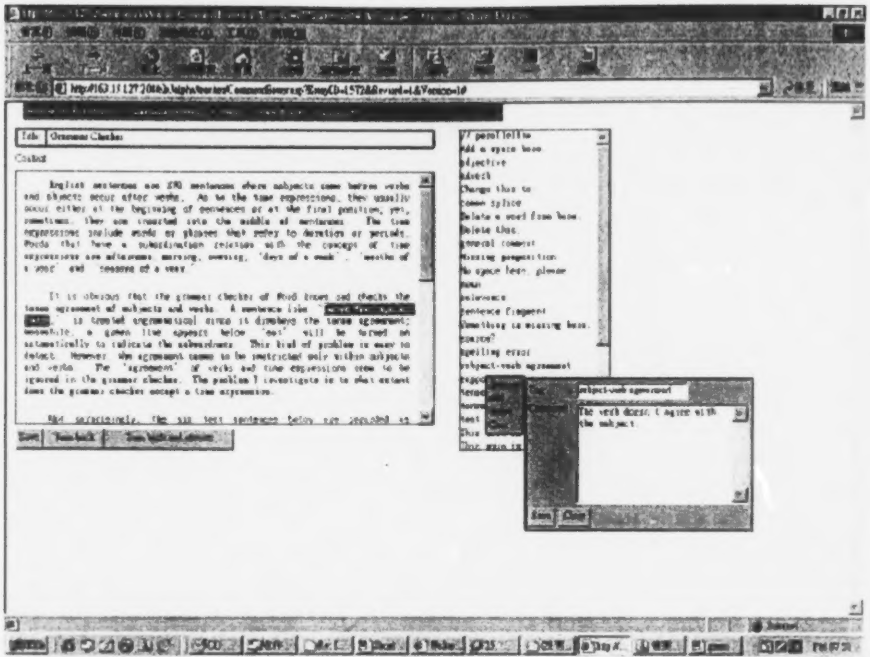
2.1 The Teacher's View

A registered teacher who logs onto the system is presented a display screen of various links to components within the system. To correct student essays, the teacher links to a page which displays their student roster.

From this roster screen, the teacher retrieves the essay by clicking on the button that represents that essay on the roster page. To mark the essay with a comment (for example, to mark a run-on sentence or subject-verb agreement problem), the teacher first chooses the portion of the essay targeted for comment using the mouse. Once the relevant text has been selected, there are two ways for the teacher to provide the student with a comment on it. The first is to simply type the comment in the empty text box provided especially for the teacher's comment and then, once the comment has been composed, append it to the intended portion of the student's text by clicking on the appropriate button ('Give the comment'). The second way of providing a comment is to choose one that has been stored in a "Comment Bank." This second way deserves some elaboration.

The Comment Bank provides each teacher with a convenient means for storing and reusing frequently used comments. To retrieve a stored comment and append it to the portion of the student essay, the teacher simply selects that comment from a drop-down menu and clicks on it. The teacher can add new comments to her Comment Bank at any time. (See Figure 1)

At this stage, research is needed to understand the factors effecting how beneficial various sorts of comments are in helping students with their writing. An advantage of this system is that, with it, researchers can control the crucial variables (such as the precise form and content of the teacher feedback being investigated), and it makes readily available the data needed for such research since the marked and unmarked essays are archived in forms that can be queried. Moreover, the revised versions of an essay can be examined along side the teacher's comments that were given to the student on the original version of the essay, making it possible to easily track the influence of various types of teacher feedback.



The verb doesn't agree with the subject.

Figure 1

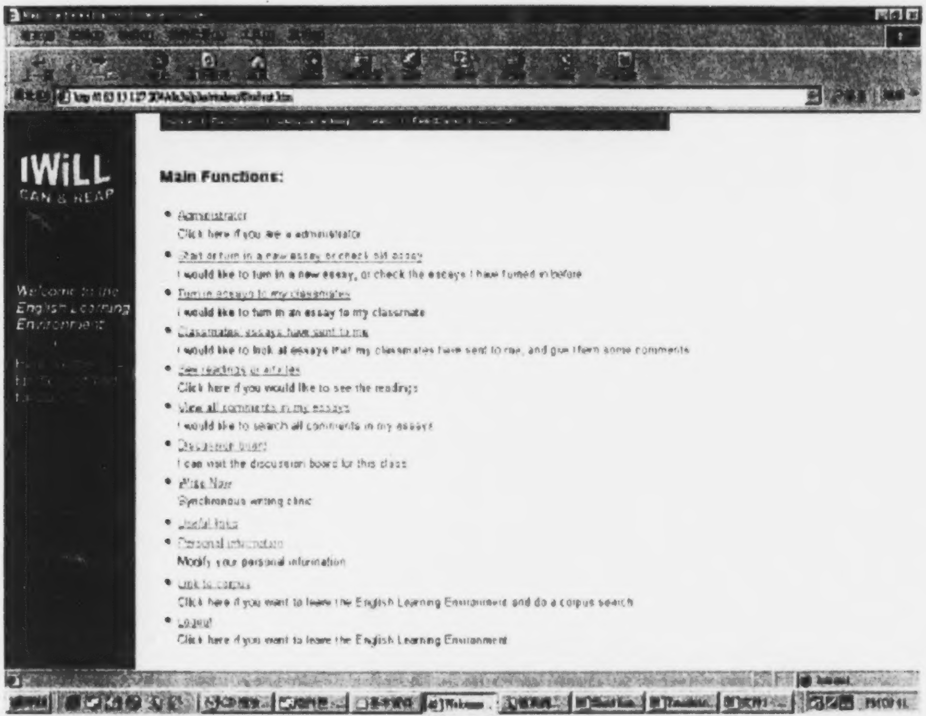
It is important to notice the distinction between this essay-marking function and the superficially similar functions offered in commercial word processors such as Microsoft Word. Like our system, those programs allow users to select portions of text and annotate them with comments. While the convenience that this provides to users as a communication tool is essentially the heart of the function's role in these commercial word processing systems, in our system this convenience is a relatively incidental (though valuable) advantage. For us, the substantial value comes from a set of related features which the word processing programs do not offer. Specifically, all of the annotations provided in our system by teachers when they mark essays are permanently indexed, by way of database technology, to the portions of text that the teacher has marked. Moreover, the essays themselves along with the indexed teacher comments enter a permanent corpus of learner essays that can be searched on line. Information extraction techniques, then, make it possible to provide learners and teachers with instant cumulative profiles of the trouble spots of individual learners, of whole classes of learners, or subtypes of learners selected by a wide variety of criteria. For example, the system enables teachers to retrieve all tokens that have been marked with a particular error type either from the essays of a single learner or from the essays of groups of learners. Moreover, teachers can retrieve the tokens of every error type and display them in order of frequency, with the error type that has been marked on the highest number of text portions listed first.

The role which our commenting function can play is deepened greatly by the highly integrated nature of our system design. Not only does it support profiles of entire groups of learners, but the analysis of the common errors can be immediately used by researchers to analyze the sources of learner difficulties. This sort of data makes it possible to investigate pervasive patterns of difficulty in the learners' English (that is, to investigate what some applied linguists call the 'interlanguage' of learners). Results of such analyses can directly enhance the entire web-based writing platform. Specifically, we have developed an authoring tool for designing online help which targets precisely the problems uncovered in the analysis of learners' errors. Moreover, based upon this sort of data, researchers can improve the design of teaching and reference materials. (See section 2.3 below for more details.)

BEST COPY AVAILABLE

2.2 The Student's View

A registered student logging onto the system is first shown a menu of links, including a link to a discussion board dedicated to the students in that class and links to helpful websites for ESL writers. To compose or turn in an essay, the student links to a page that displays a row of colored buttons, basically each button (or cluster of buttons) representing a different essay the student has written or is in the process of writing. From this page, the student can opt to resume work on an unfinished essay or revision, or to submit or compose a new essay. (See Figure 2)



Main Functions:

- Administrator
  Click here if you are a administrator
- Start or turn in a new essay or check old essay
  I would like to turn in a new essay, or check the essays I have turned in before
- Turn in essays to my classmates
  I would like to turn in an essay to my classmate
- Classmates' essays have sent to me
  I would like to look at essays that my classmates have sent to me, and give them some comments
- See readings or articles
  Click here if you would like to see the readings
- View all comments in my essays
  I would like to search all comments in my essays
- Discussion board
  I can visit the discussion board for this class
- Write Now
  Synchronous writing clinic
- Useful links
- Personal information
  Modify your personal information
- Link to corpus
  Click here if you want to leave the English Learning Environment and do a corpus search
- Logout
  Click here if you want to leave the English Learning Environment

Figure 2

To compose an essay, students can elect either to compose online by typing their essay within a designated text box on the appropriate page or to copy and paste into that box an essay composed off-line. The latter essays are imported as text files.

From this screen where the essay has been composed or imported, students can submit the essay to the instructor. Alternatively, through a drop-down list of all of their classmates' names they can send the essay to any number of their classmates for peer editing or commenting. The methods of selecting portions of text for comment and for submitting comments are basically the same as under the teacher's view described above.

When a student views an essay that has been marked by the teacher, the essay itself appears almost identical to the student's original, unmarked essay. None of the teacher's comments are immediately visible. The only difference in the appearance of the marked and unmarked version of an essay is that in the 'corrected' version some of the student's text shows up in blue. These are portions of the essay that the teacher has marked for comment. To see the content of the teacher's comment, the student places the cursor on the blue text and the comment appears.

An important feature offered to students is a specific sort of search function which they can access through a link labeled: "Search all comments in my essays." With this function, a student can access a list of all of the comments that the teacher has marked on his essays. The comments are listed in descending order of frequency as they occur in the entire set of that particular student' s essays.

By clicking on the View button for any of these comments that appear on this display, the student retrieves a cumulative listing of all of the instances where this comment appeared in his own essays. To give the minimum context that would allow the student to see the nature of the marked problem, this search function retrieves complete sentences from the student' s texts even if the teacher had marked only a word or phrase or other proper subpart of that sentence for comment. In instances where the teacher has marked off a chunk of text which spans a sentence boundary in the student essay, the entire text of both (or all) of those sentences is displayed. By clicking on any of the tokens that have been retrieved, the student links to the complete text of the essay from which that token was extracted, thus accessing the full context.

What the "View Comments" function provides is the opportunity for the student to see patterns of difficulty, to see in one glance a set of tokens of one type of difficulty from his own writing. Of course, what is needed here is research on the differential effects of the two approaches to providing feedback. Moreover, the effectiveness of the View Comments function will almost certainly depend not simply on the fact that the system allows searches of the essays according to teacher comments, but also upon the quality and clarity of the comments themselves. An important property designed into the system is that it can track precisely the kinds of data needed for investigating these sorts of issues.

2.3 The Researcher' s View

The system has been designed to create a corpus of student essays as a byproduct of the teacher-student interaction on the system. Specifically, each essay that a student submits to his teacher over this system is, with the permission of that student, copied into a corpus of "learner English." Consequently the corpus itself grows as the system is used by students and their teachers.

The creation and analysis of corpora of learner language data is an extremely new and promising field of research (see Granger 1998). One of the formidable obstacles in this field is a practical one of how to input the learner data. Granger (1998:11) mentions three methods, all extremely tedious, time-consuming, expensive and the first two prone to error: (1) scanning essays from hard copies and (2) keying in data manually (3) downloading electronic data. Granger implies that the latter refers to collecting student essays that are on disks. Our system offers another way of creating learner corpora which goes a long way toward eliminating these prohibitive drawbacks. The texts created by students enter the corpus virtually unaffected by any intermediate steps for "inputting" them because the exact text that the student sends to the teacher over the system is copied into the corpus. Moreover, when students first register to use the system, they provide relevant metadata about their years of studying English, their gender, age, mother tongue, and the relevant fields of metadata are updated every semester. Each essay a student turns in is automatically indexed to this information and annotated with the date when that specific essay was submitted. This indexing allows for longitudinal studies of learner writing as well as cross-sectional studies that consider variables such as gender, age, or years of study. Researchers can add other fields of metadata to track other variables for specific studies.

Researchers are not only able to search the corpus of essays collected from learners. The results of the researcher' s analyses of learner difficulties can be translated into the content of an active online help function for those learners. The system includes an authoring environment for content administrators (ICPs) where they simply indicate what string of text in a learners essay should trigger help, and then write the content of the help which should be displayed for that particular string. Research on the learner corpus has revealed, for example, that the word 'ever' was misused by learners in 25% of the cases where it appeared in their essays. Further analysis attributed this to negative transfer in which learners associated the English expression 'ever' with a Chinese counterpart expression (*cheng jing*). These two expressions while overlapping in use and meaning, diverge in important ways, and it is precisely in these diverging respects where students misused the English expression. Based on this linguistic analysis, the authoring environment for online help was used to design advice concerning the word 'ever' addressing precisely the difficulties it poses for Chinese learners. When learners request general help on an essay, the help function actively detects instances of 'ever', highlights them and creates a link to this advise.

3 Conclusion

The underlying goal of the project described above has been not only to create an online writing environment that connects teachers and students by way of a user-friendly interface, but also to provide ways to exploit the valuable data that is created when the environment is used. The learners' essays themselves are stored in growing corpus of ESL language production. The comments that teachers append to the particular segments of the learners' texts in the course of essay correction are treated as annotations of those texts, which can be searched and retrieved. An authoring environment for online help permits content administrators to turn interlanguage research results into highly specific help concerning attested difficulties which traditional language education has neglected. It is hoped that increasingly sophisticated and dynamic manipulations of these sorts of data will lead to the delivery of evermore useful and useable information to learners, teachers, and researchers both online and off.

References

[1] Chang, Hsiu-Chen, "Web-based EFL learning and teaching" (manuscript), (2000).

[2] Ferris, Dana, "The case for grammar correction in L2 writing classes: a response to Truscott" ,(1996). *Journal of Second Language Writing* 8-1:1-11, (1999).

[3] Granger, Sylviane, "The computer learner corpus: a versatile new source of data in SLA research.", In Sylviane Granger, *Learner English on Computer*. New York: Longman, (1998).

[4] James, Carl, "Errors in Language Learning and Use.", New York: Longman, (1998).

[5] Kearsley, G., "Computer-aided instruction, intelligent", In S. Shapiro (ed.) pp. 154-9, (1987).

[6] Kuo, C.-H., David Wible, and Chia-Lin Chio, "On the design of a synchronous virtual writing clinic." (manuscript), (2000).

[7] Kuo, C.-H., David Wible, Meng-Chang Chen, Li-Chun Sung, Nai-Lung Tsao, Chia-Lin Chio, and Chih-Chiang Wang "The design and implementation of an intelligent web-based interactive language learning system." (manuscript), (2000).

[8] Krashen, S., "The input hypothesis and its rivals", in Ellis, N. (ed.) *Implicit and Explicit Learning of Languages*, London: Academic Press, (1994).

[9] Ogata, Hiroaki, Yano, Yoneo, and Wakita, Riko CCML —"Exchanging marked-up documents in a networked writing classroom.", *Computer Assisted Language Learning*, 11-2:201-214, (1998).

[10] Nerbonne, John (ed), "*Linguistic Databases.* Stanford": CSLI, (1998).

[11] Neuwirth, M.C. and Wojahn, G.P., "Learning to write: computer support for a cooperative process.", In T. Koshmann, CSCL: Theory and Practice of an Emerging Paradigm. Lawrence Erlbaum Associates, pp.147-70, (1996).

[12] Selinker, L., "*Rediscovering interlanguage.*", London: Longman, (1992).

[13] Selinker, L., "On the notion 'IL competence' in early SLA research: an aid to understanding some baffling current issues.", In Brown, G., Malmkjaer, K. & Williams, J. (eds.). *Performance and competence in second language acquisition.*, Cambridge: Cambridge University Press, 92-113, (1996).

[14] Truscott, John, "The case against grammar correction in L2 writing classes.", *Language Learning* 46-2:327-369, (1996).

Internet Video on Demand System of Classroom Teaching Cases

- Building "RAPSODY": An Intelligent Media-Oriented Remote Educational System for Self-Learning Support -

Hisayoshi INOUE and Toshio OKAMOTO
Graduate School of Information Systems
The University of Electro-Communications
1-5-1 Chofugaoka, Chofu-City, Tokyo, 182-8585 JAPAN
TEL: +81-424-43-5623
E-mail: {ino,okamoto}@ai.is.uec.ac.jp

Our study aims to accumulate information for teachers, about established teaching methodologies and techniques. For the purpose of our study, we construct a searching system for lesson example videos, and we develop an environment for facilitating the usage of our example videos, and for encouraging teachers' collaboration. Concretely, we focus on the domain of the new subject introduced in the Japanese curriculum, called "Information Education". In this paper, we describe the details of our video searching system, the design of the database tables, and we show an example of system operation.

Keywords: **Information Education, Teacher Education, Distance Learning, VOD, Rapsody**

1 Introduction

In now-a-days advanced information society, the demands about teachers' competence are high and diverse. Concretely, teachers are required to posses on one hand curriculum development ability, learning environment design ability, group learning, individual learning, simultaneous learning, and on the other hand the previous abilities should be supported by class practice ability, observation and evaluation ability, and last but not least, the ability to connect the textbook's world and the real world.

It is difficult to raise and form this complex set of abilities, with the help of only the presently available education and training methodology for teachers. Therefore, the necessity emerged [1] to examine the feasibility of a new systematic approach, for supporting the teachers' literacy progress, by building on their natural talents/ and abilities, and expanding these to reach the required width and breadth.

The information technology science is offering the tools for the development of an environment supporting the teachers' endowment progress. The knowledge concerning the teachers' education contents and methods is stored as multimedia information, in the form of pictures, videos and sound tracks. Moreover, by using the network environment, it is possible to make use of all resources over the net, without any constraints or restrictions of time and/or geographical location.

With this goal in mind, we are researching the development of an integrative distance education training system for supporting teachers' self-training, called RAPSODY (Remote and Adaptive System of Oriented Dynamic Teaching/Learning) . Up to now, the availability of video records and guidance plans about lessons was limited to education training centers or universities, etc. The present research intends to make the information on educational activities and practices public, and aims at joint usage and re-usage of teachers' self-learning and self-training methodologies and tools. Concretely, we develop a retrieval system based on dialogue patterns, by using a database of lesson videos.In order to jointly use the information in the distributed environment, or to be able to re-use it, we implemented a Video On Demand (VOD) system. The (teacher) user can control/manage the specification and stop/suspension of the regeneration point for the distributed VOD.

The main purpose of our research is therefore to propose a distance-learning environment on the Internet, for improving the teachers' practical abilities. In this paper, we describe the video on demand system developed until now, the indexing method of the classroom teaching movie example database, the system's functions and the system's evaluation.

2 The outline of the system

2.1 The structure of the system

Fig. 1 shows the structure of the search system. The system is built of the following three parts:

- Web browser;
- Lesson video example database;
- Video distribution server.

The web browser has the role of the user interface. The search/retrieval mechanism searches the lesson example video database via three types of relational database files.

The video distribution server stores the lesson scenes' videos. The video distribution server performs the VOD function at the users' requests. The search functions performed for the user are of the following two types.

- Keyword Search
- Feature Oriented Search

The Keyword Search (fig. 1,) takes place as explained below.

a-1) The user designates the search conditions.
b-1) The search mechanism compares the search conditions input by the user, with the available class example video database.
c-1) The search mechanism extracts the record(s) matching the searching conditions.
d-1) The result is displayed as the search result.

The Feature Oriented Search process (fig. 1,) is done as shown in the following.

a-2) The system dialogue mechanism enquires about the video characteristics/features desired by the user.
b-2) The user can answer to the system's enquiry vaguely [2].
c-2) The decision making table (showed later on in table 6), obtained from the user, is the basis for the evaluation of the specific search conditions. The gathered search conditions are passed over to the search mechanism of the database.
d-2) The database search mechanism compares the search conditions resulting from the evaluation with the lesson example videos contained in the database.
e-2) This result is displayed as the search result.

The search result is formatted as an URL list that is shown to the user. These URLs perform the function of linking the search result and the actual videos on the VOD server. The (teacher) user chooses the URL that s/he wants to refer (Fig. 1,).When the URL is chosen, the VOD client software, embedded via the Web browser plug-in, starts, and the video playback begins.

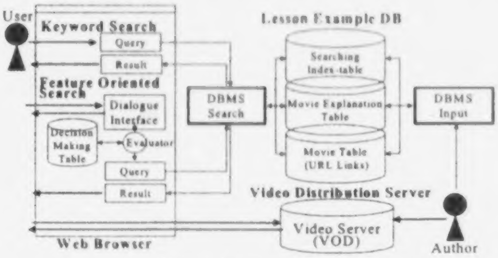

User
Keyword Search
Query
Result
Feature Oriented Search
Dialogue Interface
Decision Making Table
Evaluator
Query
Result
Web Browser
DBMS Search
Lesson Example DB
Searching Index-table
Movie Explanation Table
Movie Table (URL Links)
DBMS Input
Video Distribution Server
Video Server (VOD)
Author

Figure 2 Sstem Organization

3. Database structure

3.1 Lesson (unit) database

The following three relational database files define the video lesson database.

- Searching Index File
- Movie Explanation File
- Movie File

In the following, we will explain in detail each database file type.

3.2 Searching Index File

The Searching Index File results from the comparison of the video database with the search items. The search items are organized in items for the Keyword Search and items for the Feature Oriented Search. In table 1, we show the Searching Index File for the Keyword Search, and in table 2, the Searching Index File for the Feature Oriented Search.

For the Keyword Search, the search conditions are given directly by the user. The Searching Index File slots are "lesson name (unit)", "learning contents (subunit)", "used information equipment/machinery/device(s)", "used tool(s) application(s)" and "class viewpoint".

In the Feature Oriented Search, the system generates the search conditions, based on the information obtained from the user. The Searching Index File (Feature Oriented Search) employs comparison of the extracted search conditions and the database, depending on the dialogue with the user. The slots of the Searching Index File for the Feature Oriented Search mechanism are "teacher activity " and "student activity ".

Table 1 Searching Index File (Keyword Search)

index frame database basic search key

record fields	value type
ID (primary key)	*numerical value type*
lesson name (unit)	*text type (menu selection)*
learning contents (subunit)	*text type (menu selection)*
used information equipment/ machinery/ device(s)	*text type (menu selection)*
used tool(s) application(s)	*text type (menu selection)*
class viewpoint	*text type (menu selection)*

Table 2 Searching Index File (Feature Oriented Search)

index frame database feature search key

record fields	value type
ID (primary key)	*numerical value type*
teacher activity	*text type (menu selection)*
student activity	*text type (menu selection)*

3.3 Movie Explanation File

Table 3 shows the contents of the Movie Explanation File, regarding the movie features. When the user is about to commence the lesson, the points, which need his/her attention, are explained via the contents of the Movie Explanation File. These explanations are used when displaying the search results.

Table 3 movie explanation file

movie feature description video feature information)

record fields	value type
camera angle	*text type (menu selection)*
equipment existence	*text type (menu selection)*
equipment usage	*text type (menu selection)*
number of teachers	*text type (menu selection)*
teachers' movements	*text type (menu selection)*
existence learning supporter(s)	*text type (menu selection)*

number of students	*text type (menu selection)*
students' movements	*text type (menu selection)*
existence of a central student	*text type (menu selection)*

The explanation information in the Movie Explanation File (table 4) resumes the lesson scenes compiled by the video registrants, and the information on how the checkpoints, necessary for the lesson, were estimated. The slot of table 4 called "teacher's aim", corresponds, for instance, to the classification 8 presented later on in table 7.The "checkpoints 1 to 3" express the free description of the image, from the points of view shown below.

Checkpoint 1 the movie preconditions to be considered;

Checkpoint 2 what should be extracted/ understood from the current movie;

Checkpoint 3 the necessary forecast of the movie's following development.

Table explanation file
4 movie

movie explanation database

record fields	value type
ID (primary key)	*numerical value type*
teacher's aim	*text type (menu selection)*
checkpoint 1	*text type (item description · within 100 characters)*
checkpoint 2	*text type (item description · within 100 characters)*
checkpoint 3	*text type (item description · within 100 characters)*

3.4 Movie File

Table 5 shows the Movie File. The Movie File contains pointers to the real videos. The VOD server houses the real videos. Table 5 contains the Movie File slots called "thumbnail picture (still picture)", "previous movie", "movie URL (movie file name)", and "next movie".

Table 5 movie file

movie database

record fields	value type
ID (primary key)	*numerical value type*
thumbnail picture (still picture)	*text type (still movie file name)*
previous movie	*text type (URL type input)*
movie URL (movie file name)	*text type (URL type input)*
next movie	*text type (URL type input)*

For the discrete movie time-series {P(t)}, the following relationship exists:
$[P(t-1),P(t),P(t+1)]$ = [previous movie, movie URL, next movie];$P(t=0)$ = {still picture};where t is the time.

4 The system's behavior

Figure 2 shows the search conditions input interface (for Keyword Search). Figure 3 shows the search result display interface. After the (teacher) user specifies the conditions for the desired video search via the search conditions input interface, the search starts. The result of this is displayed in the search result display interface [3] [4]. This interface shows the value of the slots called " still picture", "lesson contents (subunit)", "teacher's aim", "checkpoints", "teachers' activity" and "students' activity". The "still picture" can be seen in figure 3 (). Next to being a significant snapshot of the lesson video, the still picture has also the role of a pointer to the real video (a link to the VOD video file), so describes the URL (figure 3,). By clicking on the still picture, the video starts (figure 3,). The "teacher's aim" (fig. 3,) and "checkpoints" (fig. 3,) are, as mentioned before, the most important information for image explanation. The figure also presents the (teacher) user with help/support information about other items and record fields.

BEST COPY AVAILABLE

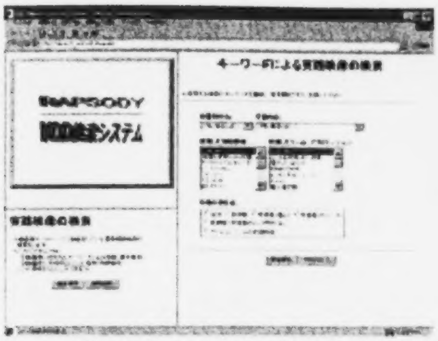


Figure 2 search conditions input picture

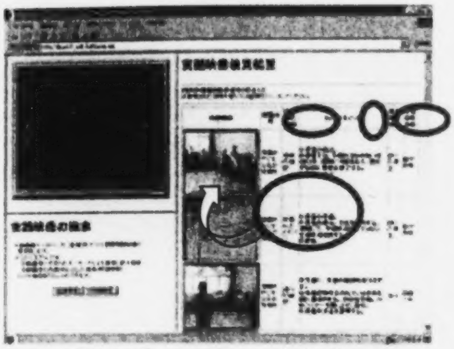


Figure 3 search result display picture

5 Conclusions

We construct a searching system for lesson example videos, and we develop an environment for facilitating the usage of our example videos, and for encouraging teachers' collaboration. Concretely, we focus on the domain of the new subject introduced in the Japanese curriculum, called "Information Education".We have presented in this paper the summary of the video search VOD system we have developed, moreover, we have shown the database organization and the system's behavior.As for the future tasks and research, we are planning to investigate about building a flexible key for the video search mechanism. We are studying at present the dialogue mechanism, with the immediate goal of using the search result's negative feedback information to the user's request, to serve as a new search key.

References

[1] Okamoto, T. (research representative/delegate) : "1998. Advanced Information Society; synthetic research on the contents, system and form of teachers' education – midway report", 1997, 9 year research funds foundation of the Ministry of Education, A)(1)09308004 stage report.

[2] Stain, A, Gulla, J.A, Muller, A. and Thiel, U.: "Conversational Interaction for Semantic Access to Multimedia Information", Intelligent Multimedia Information Retrieval, pp. 399-421, MIT Press, 1997.

[3] Noguchi, H.: "Data representation and management in image databases", IPSJ, Vol.33, No.5, pp. 457-465, 1992

KNOWLEDGE ANALYSIS OF TASKS FOR COURSEWARE DESIGN

Lorna Uden & Neil Willis
School of Computing, Staffordshire University,
The Octagon, Beaconside, Stafford, ST18 0AD, UK.
Phone: +44 (0)1785 353276
Fax: +44 (0)1785 353497
E-mail: L.Uden@staffs.ac.uk
E-mail: N.Willis@staffs.ac.uk

Substantial changes have taken place in the nature of jobs over the last twenty years, primarily due to the proliferation of computer-based technology. These changes have shifted the demands on human performance from primarily physical to cognitive. Cognitive tasks components usually have significant 'transfer task' components in addition to procedural components. Task analysis is a critical step in the development of computer-assisted instruction. Traditionally there are two well-known methods for analysing tasks in courseware applications. These are Hierarchical Task Analysis and Information Processing Task Analysis [3]. Although both of these methods have been effectively used in designing training for procedural skills, they offer little insight for analysing training involving complex cognitive tasks. Both of these analyses identify the external structure of the information, but do so independently of how they are actually structured in human memory. We believe that Knowledge Analysis of Tasks (KAT) offers an alternative method for task analysis in courseware applications. KAT differs from other task analysis methods in that it is based on the theory that people not only perform tasks, but also develop structures to represent the knowledge that they require to perform a particular task. The results of KAT are used to produce a model of tasks in terms of Task Knowledge Structures (TKSs). TKSs represent the knowledge people possess about tasks they previously learned and performed in a given dimension. The application of the theory of TKS and the analysis method of KAT can be used effectively in the design of interactive courseware applications. This paper describes a case study showing how KAT can be applied in the task analysis of interactive courseware applications.

KEYWORDS: **Task Analysis, Task Knowledge Structures, Knowledge Analysis of Tasks, Task Procedures.**

1 INTRODUCTION

According to Jonassen, Hannum and Tessmer [6], task analysis is the most crucial part of the instructional design process. Task analysis solves three main problems for designers. These are:

- It defines the content required to solve the performance problems or alleviate a performance need.
- The process forces the subject matter expert to work through each instructional step.
- It helps the designer to view the content from the learner's perspective.

There are several typical methods of conducting task analysis in courseware design. The most prominent methods are the Hierarchical Task Analysis and Information Processing Task Analysis [3]. However, the main weakness of these methods is that they are only loosely connected to any theory of psychology that would allow the task analysis to result in principles and theoretically based recommendations. Task analysis is also a critical part of human-computer interaction design. A variety of techniques are available for performing task analysis in user interface design [10]. However, the theoretical or empirical evidence

BEST COPY AVAILABLE

offered by these methods for their existence is weak [7].

We believe that Knowledge Analysis of Tasks (KAT) offers an answer to these limitations. KAT differs from other task analysis methods in that it is based on Task Knowledge Structures (TKS), which are an attempt to provide a theoretical underpinning to the method. KAT is concerned with assessing and modelling the knowledge people possess and utilise in carrying out tasks. It identifies the knowledge requirements of tasks and is aimed at assisting in the generation of design solutions. KAT has been used successfully in designing effective software systems [8]. It is our belief that KAT can offer an effective task analysis method for courseware design. This paper describes a case study showing how KAT can be used to analyse tasks for searching for information on the Internet in a courseware application. Section two of the paper presents a brief review of KAT and TKS. This is followed by the case study of the courseware application. Section four concludes the paper with recommendations for further research in KAT and TKS in courseware applications.

2 Overview of KAT & TKS

Johnson and Johnson [7] developed the Knowledge Analysis of Tasks (KAT) method. It incorporates Task Knowledge Structures into the process of task analysis. Johnson and Johnson [7] describe KAT as the collection of data from a variety of sources, such as task performers, instructors, supervisors and others, directly or indirectly affected by the task. The results from this analysis are used to produce a model of tasks in terms of task knowledge structures.

2.1 Task Knowledge Structures

According to Johnson and Johnson [7], task knowledge is represented in a person's memory and can be described by a Task Knowledge Structure (TKS). TKSs are assumed to be acquired through learning and previous task performances, and are dynamically represented in memory. This is akin to the theoretical position taken by Schank [11] in assuming that the knowledge of frequently occurring events is structured into meaningful units in memory. Empirical support for this assumption can be found in the work of Galambos [4], who showed that people recognise and use structures of events, such as the order, the sequence and importance of activities within the event sequence, to understand, explain, and make predictions about these events. Another support for this is the work of Graesser & Clark [5] on text comprehension. The assumption is that all the knowledge a person possesses about a task is contained within the TKS and that the TKS is activated in association with task performance. The suitability of utilising this form of conceptual knowledge for teaching a computerised domain is supported by Scumalhofer and Khun [12], who state that mental models present a more complete picture of the cognitive components in learning to use a computer system.

2.1.1 Implications of TKS

If we assume the existence of task knowledge structures, by which people structure their knowledge in a particular way, it follows that this task knowledge can be analysed, modelled and predicted [7]. This has important implications for learning. It suggests that by understanding the knowledge structure, recall and processing could be optimised to give quick and efficient task performance by appropriate training techniques and interface design. According to Johnson and Johnson [7], people acquire knowledge about tasks and subsequently transfer this knowledge to new or different tasks. Therefore usability and learnability are directly related to the amount of knowledge that the person is able to transfer from one task to another. The benefits to be gained from enhancing this transfer come in the form of reduced time and the achievement of a higher level of task performance in a shorter length of time.

A TKS is related to other TKSs by a number of different relations, which include temporal or experimental relations. TKS theory provides a method for the analysis and modelling of the tasks in terms of goals, procedures, actions and objects. In addition, TKS theory identifies the representitiveness (typicality) and centrality (importance) of a particular aspect of task knowledge. Within each TKS, different types of knowledge are represented. There are four components to a complete TKS model. These are as follows:

1. A goal-oriented substructure.
2. Task procedures.
3. A taxonomic substructure comprising the generic task actions and objects.
4. A summary task knowledge structure.

Task models in terms of TKS involve goal-oriented and taxonomic substructures and procedures. A goal-oriented substructure can be represented by a network of structured goal nodes that direct sequences of events that unfold over time and eventually satisfy subgoal nodes. Goal nodes can vary in hierarchical level. Goals and subgoals can be represented by nodes with links between them. Nodes can be treated as conditions, as states or as desired states (subgoals). Subgoals can also be hierarchically and concurrently related to each other. The goal-oriented substructure ' calls up' appropriate knowledge from the taxonomic substructure by use of procedures. Associated with subgoals are sets of procedures that have to be executed in order to achieve subgoals directly or indirectly. Any subgoals may give rise to further planning activity and subsequent subgoals and this may be indirectly related to a procedure set.

Task procedures define the of action object combinations in the execution of a given subgoal. They are the processes by which the taxonomic substructure is activated. The taxonomic substructure represents knowledge about generic actions and objects and the relationships between them.

We have chosen TKS as our approach because we believe that the theoretical or empirical evidence for their existence, offered by other cognitive task analysis methods is weak. Although production rules of Anderson [1] can be used to model goals, operations, methods and selection rules, we believe that these production rules also bear a closer relationship to the way a person does (or would) structure their knowledge of task [7]. We also believe that the knowledge a person possesses about a task is contained with the TKS and that the TKS is activated in association with the performance. The suitability of utilising this form of conceptual knowledge for teaching a computerised domain, is supported by Scumalhofer & Kuhn [12], who stated that mental models present a more complete picture of the cognitive complexities in learning to use a computer system. If we assume the existence of task knowledge structures, by which people structure their knowledge in a particular way, it follows that this task knowledge can be analysed, modelled and predicted [7]. This line of thought has great implications for the study of learnability. It suggests that by understanding the knowledge structure, recall and understanding can be optimised to give quick and efficient task performances, via appropriate training techniques and interface design. Johnson and Johnson [7] suggest that usability and learnability are directly related to the amount of knowledge a person is able to transfer from one task to another. The benefits to be gained from enhancing this transfer come in the form of reduced training and the achievement of a higher level of task performance in a shorter length of time.

2.2 KAT Method

KAT can be divided into three parts: knowledge gathering, knowledge analysis and TKS construction. Before the analysis is undertaken, it is imperative that the objective of the KAT exercise be identified. This is to establish domain boundaries and help to ensure that attention is concentrated on the most critical and relevant activities. Once this has been accomplished, domain gathering can begin. The principle inputs to this stage are the objective itself and the information gathered through interviews, questionnaires and observations. The major output is a preliminary picture of the domain knowledge expressed in terms of procedures, actions, objects and goals. The next stage, knowledge classification, takes the outputs of the first stage and categorises each in terms of its representative, central and generic properties. During the final stage, the actual TKS model is constructed.

3 Case Study

The case study we have chosen to demonstrate KAT is "How to use the Internet". The purpose of the task analysis is to identify the tasks or domain the students have to learn in order to use the Internet effectively. Stage 1 of KAT is Knowledge Gathering.

3.1 Knowledge gathering

A wide range of powerful methods of cognitive task analysis have been developed and applied over the last ten years [9]. Although these methods have been used successful in many applications, few have become accessible to training practitioners and instructional designers. All of these methods required considerate time and resources. All have been part of research effort conducted by scientists as opposed by practitioners. We have chosen to use Applied Cognitive Task Analysis (ACTA), which is a streamlined cognitive task analysis method, developed by Klein Associates [9], We have chosen it as our method to elicit task knowledge from our subject matter experts. The main reason for our choice is that ACTA allows us to elicit and represent cognitive components of skilled task performance and the means to transform those data into design recommendations without having to be knowledge engineers, cognitive psychologists and human factor/ergonomics professionals.

We used a structured interview for eliciting task knowledge from our subject-matter experts. The method we used is known as). ACTA consists of four sequential steps, of which the first three steps comprise three different stages of interviews. The final step is used to sort and organise the gathered information into what could be the most important tasks. The four steps are Task Analysis Interview, Knowledge Audit, Simulation Interview and Cognitive Demands. It is beyond the scope of this paper to describe these steps. Interested readers are invited to contact the first author.

3.2 Analysing the Task Knowledge

The data collected through the ACTA interviews formed the basis of the data for the knowledge analysis of tasks. It provides information about the actions and objects and the goals and subgoals that are involved in a task. The simulation interviews also provided opportunities to find out information about the procedures and strategies used in performing a task.

The knowledge gathered during the ACTA process needs to be identified in terms of the components of a TKS, namely actions and objects, and the structure of those objects, procedures and goal structure. In addition, the knowledge gathered and identified has to be analysed in order to identify representative, central and generic properties of tasks within a given domain or across domains. Some task components are more representative/typical of a task than are others. Central task components are those necessary to successful task execution. Without these central components, the task goal will fail to be achieved. Generic task components, on the other hand, are those common across a number of task performers.

3.3 Task Model Construction

Having gathered and identified the knowledge required for the tasks to be learned, the next stage is the construction of the task model using TKS. The aim of this approach is to identify the structure, content and attributes of learners' task knowledge in line with TKS theory. There are three components to complete a TKS model:

- Goal-oriented substructure
- Task procedures
- A taxonomic substructure from the generic task actions and objects.

3.3.1Goal-oriented sub-structures

The goal structure component of the model identifies the relations between different goal states. There are two general forms of relations: hierarchical and control relations. Hierarchical relations show how a goal can be decomposed into further collections of sub-goals. Control relations show how goals and sub-goals are related to each other for execution. They include sequential, parallel, unordered and optimal relations. Both hierarchical and control relations can appear in all parts of the goal structure components of the TKS model. Goal structures can be represented diagrammatically using tree structures to represent hierarchical components, and transition networks to represent control relations. Figure 1 shows a tree diagram of the evaluation of a search result.

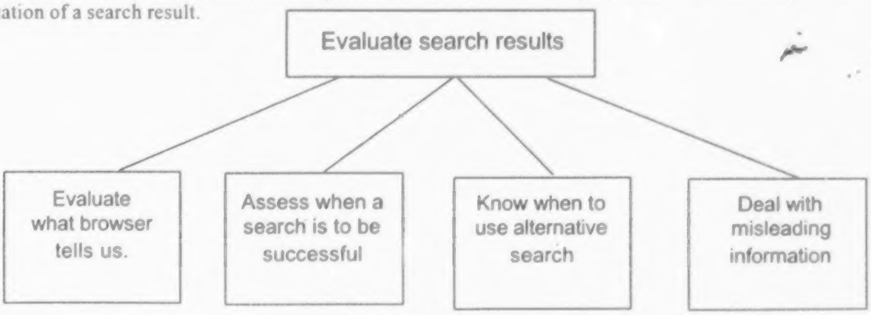

Evaluate search results
Evaluate what browser tells us.
Assess when a search is to be successful
Know when to use alternative search
Deal with misleading information

Figure 1 A Tree diagram for Evaluation of Search Result.

3.3.2 Task Procedures

These are directly related to the lowest level goals in the tree structure and they represent the detailed executable form of the task. Task procedures contain actions and objects and the relations between them. The relations include sequential, parallel, iterative and conditional control relations. These control relations exist within a procedure body and determine how the actions will be executed with respect to the objects. Procedures can 'call' other procedures and form parts of the bodies of other procedures like programming languages. Each procedure is defined by a precondition, which determines the context that must exist before it can be executed. Upon execution of a procedure, a defined post-condition will result. The pre and post conditions of procedures are defined in the procedural sub-structure and provide a way of representing the relationship between the goal structure and the procedural sub-structure.

Task procedures are the process by which the taxonomic sub-structure is activated. Not only that, the tasks may be decomposed in different ways; there may also be a choice between a number of different strategies which are context -dependent competing sets of procedures. One set of procedures may be more appropriate than other sets. Contextual information and the circumstances under which the task is to be executed will affect strategy appropriateness. Single procedures in a given strategy may differ in how central they are to the task as a whole. The task procedures for "Deal with misleading information and instruction" are shown in Figure 2. Included in the task procedures are actions and objects of tasks.

```
START
1.0       Knowing How to Deal with Misleading Info & Instructions
1.1       IF results returned matches, THEN
1.1.1                       IF the most relevant link can be found, THEN
1.1.1.1                          Click on the link to go to the page
1.1.1.2                          IF NOT FOUND message is displayed (i.e. Link is broken), THEN
1.1.1.2.1                             Apply Edit-URL strategy process to find page
                                 ENDIF
                            ENDIF
          ELSE
1.2                         IF results did not return no match, THEN
1.2.1                            Modify search strategy
1.2.2                            Perform search process
1.2.3                            Go to step 1.1
                            ENDIF
          ENDIF
END
-----------------------------------------------

START
2.0       Edit-URL strategy process
            DO
2.1                         Position cursor at the end of the URL in the Address bar
2.2                         Start deleting the last part of the URL and stop before the first '/'
2.3                         Perform search
2.4                         IF directory page appears, THEN
2.4.1     Browse directory to find any possible relevant link
2.4.2     IF a possible relevant link is found, THEN
2.4.2.1                   Click on the link to go to the page
2.4.2.2                   IF the desired information is not found on the page, THEN
2.4.2.2.1                                   Go to step 2.1
                                 ENDIF
                          ENDIF
                            ENDIF
          ENDDO
END

-----------------------------------------------
```

START
3.0 *Search process*
3.1 Type keyword(s) into search text box
3.2 Hit the search button
END

The actions included:

- Evaluating search results
- Choosing and clicking on a relevant link
- Evaluating the relevance of a page
- Deciding on alternative course of action when a broken link is encountered
- Deciding on another search strategy
- Editing the URL (deleting the end part of the URL)

The objects included:

- Web browser
- Browser address bar
- Browser content window
- Search keyword / phrase
- Search result
- Search engine
- Cursor
- Search button
- Search text box
- Directory page
- Hyperlink
- Not Found error message

Figure 2. The procedures for ' dealing with misleading information'

3.3.3 Taxonomic sub-structure

The taxonomic substructure represents knowledge about objects and the relationships between them. There are three levels to the taxonomic substructure. The top level is the superordinate level. The basic level of the taxonomic substructure contains the objects that constitute the superordinate task category. The basic level task category represents knowledge about: (i) in which task procedure a category member is used, (ii) which other task object a category member is related to, and what that relationship is, i.e., whether the category member causes, enables, follows or is carried out in conjunction with other task objects, (iii) which actions are associated with a category member, (iv) what features or properties a category member possesses, (v) the usual circumstances under which a particular category member occurs, for example, whereabouts in the task the category member is manipulated, (vi) whether or not the object is central and (vii) a pointer or reference to the most representative instance of that object. The bottom level of the taxonomic substructure is the subordinate task category, which contains a particular instance of the type of the object represented at the basic level. The taxonomic substructure of searching the Internet is shown in Figure 3.

Generification / Typicality (T) / Centrality (C)

Generic Objects	Specific Objects
Web browser	Microsoft Internet Explorer (T) Netscape Navigator (T) Mosaic Content window Title bar Tool bar Address bar Status bar Animated logo Bookmarks (favourites)

Search keyword (C)	Single keyword Phrase
Search result	Page title Percentage of relevance Brief description of page URL of page
Search engine (C)	Search text box Search button Alta Vista Excite Lycos Infoseek Yahoo, etc.
Hyperlink	Text (T) Image (T)
Search technique (C)	Boolean Double quotes Wildcard Search-engine specific techniques Other unique strategies
Cursor	
Directory page	
Not Found error message	

Figure 3. Taxonomic Sub-structure for Task Analysis on Searching the Internet

4 Conclusion

There are many methods available for courseware designers to use for task analysis. However, the main weakness of these methods is that they lack theoretical or empirical evidence for their existence. Knowledge Analysis of Tasks can be used effectively in the analysis of tasks for courseware design, as shown in the case study. Evidence to support that task knowledge is represented in conceptual or general knowledge structures in long-term memory is found in the work of Galambos [4]. Buckley and Johnson [2] have demonstrated the usefulness of TKS in the analysis of a prototype messaging system. The main achievement of KAT has been the modelling of tasks in terms of TKS. Each component of TKS represents different types of knowledge required by the tasks, whilst the goal structure identifies relations between different goal states – both hierarchical and control-relations. These models are represented diagrammatically using tree structures for hierarchical and transition networks for control. KAT offers the courseware designer a powerful method with which to model knowledge. By modelling knowledge, the designer is forced to understand the knowledge structures with which he/she is dealing. Reflection upon these 'building blocks' of problems solving is something that other methods fail to encourage. The design is focused on the expert's cognitive skills, rather than on the technical details of design. The models in KAT also provide an excellent reference point for the designer to return to during implementation in order to refocus the design if needs be. The biggest drawback of this method is the time it takes for design. However, we believe that the benefits of KAT outweigh its drawbacks and strongly recommend its use.

REFERENCES

[1] Anderson, J. R., "Cognitive psychology and its implications", Freeman, (1985).

[2] Buckley, P. & Johnson, P. (1987). Analysis of communication tools for the design of a structured messaging system. http://www.dcs.qmw.ac.uk/publications/report_abstracts/1987/403

[3] Gagné, R.M., Briggs, L.J. & Wager, W.W., "Principles of Instructional Design Holt", Rinehart & Winston. 4th Edition, (1988).

[4] Galambos, J.A. "Knowledge Structures for Common Activities", In J.A. Galambos, R.P. Ableson & J.B. Black (eds.), Knowledge Structures, Lawrence Erlbaum Associates, (1986).

[5] Graesser, A.G. & Clarke, L.F., "Structures and Procedures of Implicit Knowledge", Ablex, (1985)

[6] Jonassen, D., Hannum, W. & Tessmer, M., "Handbook of Task Analysis Procedures", Paeger, (1989).

[7] Johnson, P. & Johnson, H. "Human Computer Interaction: Psychology, Task Analysis and Software Engineering", McGraw-Hill Book Publishing Company, (1992).

[8] Johnson, P. & Johnson, H. & Wilson, S., "Rapid Prototyping of User Interfaces Driven by Task Models," In J.M. Carroll (ed.), Scenario-Based Design: Envisioning work and technology in system development, John Wiley & Son, (1995).

[9] Militello, L.G. & Hutton, J.B., "Applied Cognitive Task Analysis (ACTA): A practitioner's Toolkit for understanding cognitive task demands", Ergonomics, Vol. 41, No. 11, 1618-1641, (1998).

[10] Roberts, D., Berry, D., Isensee, S. & Mullaly, J., "Software Engineering Series, Designing for the User with OVID: Bridging User-Interface design and software engineering", Macmillan Technical Publishing, (1998).

[11] Schank, R.C., "Dynamic Memory: A theory of reminding and learning in computers and people",. Cambridge University Press, (1982).

[12] Scumalhofer, D. & Khun, O., "The Psychological Processes of Constructing a Mental Model when learning from being told, from examples, and by explanation", In T.A. Galambos (ed.), Knowledge Structures, Lawrence Erlbaum Associates, (1986).

Making the Most of the Internet for Potential for Education

John Wang* and Dilek Sanver**
**RS Information Systems, Inc./university of California, Berkeley*
3402 Sepulveda Blvd, #421, Los Angeles, CA 90034, USA
Telephone: (805) 982-3312
Fax: (805) 982-2659
Email: jgwang@csua-berkeley.edu
***University of California, Irvine*
Irvine, CA 92697,USA
Telephone: (310)390-7429
Email: dsanver@uci.edu

Who is building Web sites today? Entrepreneurs, writers, hobbyists, educators and students from the elementary grades and up are building them, not Java programmers, In fact, very few Web sites are actually built by professional programmers That is why strategies for making the most of Internet' s potential for education is important: It brings the power of Internet to non-programming Web-builders like teachers and their students. Internet is an exciting, dynamic technology that is challenging for education. With new specifications, new classes, and general updates, one must accept the fact, when integrating Internet Technology into instruction, that the course will never be the same because the subject matter is in a never-ending state of change. In today' s technological environment, curriculum development must be iterative; in other words, it is an ongoing repetitive process that is required due to the constant change of the subject matter and the technology. In order to be making the most of Internet's potential for education, we proposed these six basic phases--- understanding, planning, research, development, refinement and implementation. This article describes how to effectively use this six-phased approach Follow these phases, the educators and learners can collaborate to enhance existing material and produce new innovations for education.

*The paper was not available by the date of printing.

Natural Language-like Knowledge Representation for Multimedia Educational Systems

Eva Heinrich and Ray Kemp
Institute of Information Sciences and Technology
Massey University, Palmerston North, New Zealand
E-mail: E.Heinrich@massey.ac.nz R.Kemp@massey.ac.nz

The appropriate use of multimedia is becoming increasingly important in computer teaching systems. Not only are students stimulated by being presented with information in a variety of forms, but such an approach also more closely resembles the real world where they have to assimilate what they see and hear, abstracting out what is relevant. With the diversity and amount of multimedia material that may be present in these systems, a powerful form of knowledge representation is required to support navigation and knowledge retrieval. The (human or computer) tutor may wish to refer to document segments, to recap important points, provide feedback, give hints and so on. The student also may wish to refer to items previously seen or heard. The Flexible Structured Coding Language, FSCL, is a natural language-like, formalised description language which allows the formulation of rich yet structured sentences. These sentences are attached to segments of multimedia documents. FSCL provides an easily accessible approach for knowledge representation, precise and rich description of complex contents, correct and complete retrieval within the descriptions, and retrieval across data of different media types. FSCL can be extended to integrate ontologies, inference of knowledge and freeform querying performed by the learner.

Keywords: **Multimedia, knowledge representation**

1 Introduction

Computer-based educational systems have developed from standalone applications, using mainly text and graphics, which focused on teaching a restricted set of subjects or skills. Today's multimedia systems are often distributed across the web using a client-server approach and aim to integrate teaching material from multiple subjects areas. These systems collect feedback on the progress of the learner and attempt to provide material at the appropriate levels. An example of such a system is GENTLE [5].

Beside the technical challenges of managing such a system, a number of conceptual ones arise. One of these is knowledge representation and the related issue of knowledge retrieval. One problem with supplying a learner with a flexible learning environment is the need to provide a mechanism for locating appropriate information. This is a non-trivial task considering the vast amount of diverse material stored and the complexity of the concepts incorporated into the learning material. Another requirement is to give the learner a mechanism for questioning the system. This can be for retrieving specific material or for asking conceptual questions concerning the subject area.

To illustrate some of the requirements for a computer-based educational system, consider a small scenario. Imagine a web-based teaching module on the use of machinery. This module could consist of a number of multimedia documents: for example, a video showing an instructor demonstrating the use of the machinery, a set of images displaying various technical features of the machinery or a set of text documents explaining various procedures. These multimedia documents, annotated with appropriate knowledge representation mechanisms and generic domain knowledge, have to be stored. Based on this information a range of material could be retrieved: a segment of the video document showing the instructor demonstrating a specific task; additional information from images or text documents relating exactly to this task; the status of

BEST COPY AVAILABLE

the machinery at a specific position in the video inferred from the domain rules.

After a brief overview on current approaches to knowledge representation in computer-based educational systems, we consider how the Flexible Structured Coding Language, FSCL [9,11], may be applied to this problem. We will first describe FSCL in the form it is used in its original context of studies of human behaviour and then discuss the advantages of using FSCL in computer-based educational systems. We then suggest some modifications to FSCL to provide extended support for computer-based educational systems and conclude the paper by summarising the contributions this natural language-like approach to knowledge representation can give us.

2 Current approaches to knowledge representation

To access the appropriate information in a computer-based educational system, a knowledge representation scheme is necessary. This provides a meta-level description of the contents of the educational system. In this paper, we consider the format of this meta-level description, not its technical realization in a database or file system. Before we describe some common approaches to meta-level description, we want to briefly discuss why a meta-level description is necessary and why it is not possible to extract the information directly from the learning material.

The retrieval of information from documents directly has limited scope both on a technical and on a conceptual level. Technically, searching through text based documents is easy and allows for identification of keywords, phrases or sentences. Achieving the same level of retrieval for video documents is much harder. Techniques exist to automatically parse video documents to detect scene changes [8, 23] and objects [6, 17]. However, a number of problems still have to be overcome to provide sufficient access to video content [13].

Setting the technical difficulties in accessing video or audio documents aside, there are still conceptual considerations which will demand some meta-level description of content. Retrieving appropriate information from a collection of documents will, in many cases, require access to the semantics of these documents. Searching through these documents on a keyword (or object) basis is unlikely to produce satisfactory results [2]. The transition of factual ('she was smiling', a smiling face, a sunny picture) to conceptual (happiness, pleasant atmosphere) information has to be made to access the semantics of a document. This is not possible without some meta information or description of these documents.

A number of approaches are used to facilitate the access to the semantics of documents in preparation for information retrieval. Ontologies provide a modelling scheme for a specific domain creating a shared vocabulary for the description of contents [4]. Topic maps [22] create organising principles for information by defining topics, the associations of topics and the occurrence of topics in documents. Conceptual graphs [20,21] capture knowledge about a specific domain and make this knowledge accessible to deduction using first order logic.

In the analysis of data in the social sciences, a *description* approach is common. Codes or annotations, called descriptions, are attached to specific locations of multimedia documents to assist retrieval. These can contain any kind of factual or semantic descriptions of the documents' contents. Domain specific codes or freeform textual annotations are common in analysis programs like The Observer [16], Nudist [18] or its successor, NVivo [19]. All the approaches mentioned above have been proposed to overcome the technical and conceptual difficulties of accessing the information contained in multimedia documents and to facilitate the retrieval of appropriate information. In this paper, we propose the use of FSCL as a meta-level description mechanism. In the next section we introduce the main features of FSCL. We follow this by a discussion of its advantages for knowledge representation and retrieval, and indicate how FSCL can be combined with ontologies and conceptual graphs.

3 Knowledge representation using FSCL

FSCL is a natural language-like description language. It aims to combine the expressiveness and flexibility of natural language with the rigour of formalised approaches. The main components of FSCL are its *vocabulary*, *grammar* and *categories*. The vocabulary can be freely defined by the author of the teaching material. Any word can be used and the vocabulary can be extended at any point of time. Whereas the vocabulary is likely to be defined for a specific domain, the grammar is generic. It is designed to formulate 'subject - verb - object' and 'concept - object' sentences and combinations of these elements, including conjunctions, prepositions, adjectives and adverbs. The role of the categories is to bridge the vocabulary and the generic grammar. The grammar is defined on the categories. Each word of the vocabulary has to belong to exactly one category. This construct allows for the structure of the description language to stay the same across applications in different domains. The categories of FSCL have been defined in accordance with the word classes of the English language. The categories are: Person/Thing, Activity, Concept, Conjunction, Preposition and Descriptor (which combines the word classes adjective and adverb).

FSCL has been incorporated into an information system to support the analysis of multimedia documents, called PAC [12]. Sentences formulated with FSCL can be, in a system like PAC, attached to a segment of a multimedia document. The sentences, together with document identifiers and segment specifications are stored in a database and later used for retrieval. Because the structure of the FSCL sentences is well known, it is possible to access the semantics of the information stored. The retrieval of information from FSCL descriptions is achieved using the Flexible Structured Query Language, FSQL [9].

FSQL provides three layers for querying: the first layer is based on the properties of FSCL and allows the correct and complete retrieval of information from the description sentences; the second layer provides for Boolean combinations within sets of description sentences; the third layer accesses the properties of the multimedia document segments attached to the FSCL sentences and facilitates time and position comparisons. More detailed information on FSCL and FSQL can be found in [9]. Specific information about information retrieval across multiple media formats is given in [10].

4 Advantages of using FSCL

The most convenient and expressive language available to us is natural language. Yet looking at knowledge retrieval with computer systems, natural language poses a range of well known and not yet fully solved problems. The main problem lies in the vast amount of implicit knowledge necessary to see words in the right context and to fully understand a sentence [21]. Various large scale projects are underway to attack these problems, like WordNet [15], an ontology for natural language processing, and the Cyc system [14], attempting to construct a 'complete' ontology of the world. Our approach is far less ambitious. We acknowledge that using full natural language for knowledge representation and retrieval would be highly desirable. Yet with the enormous difficulties associated with this approach we were looking for a much simpler solution. FSCL provides us with a number of advantages:

- We have a natural language-like notation. Any FSCL sentence can immediately be understood by a human reader. The importance of this is confirmed in the discussion of the five principles of knowledge representation by Davis *et al* [3].

- We have a language and can deduce the structure of our sentences. We have therefore more power than with the keyword approach commonly used in information retrieval, which suffers from low precision and low recall [21].

- We can build a powerful vocabulary by integrating the FSCL categories with ontologies.

- Of special interest to computer-based educational systems is that we can link our form of knowledge representation with multimedia documents.

FSCL has been successfully used to support the study of behaviour recorded in multimedia documents. It has given analysts the possibility to create rich descriptions of behaviour and to analyse the descriptions in a precise way [9]. We want to keep the main features of FSCL in formulating natural language-like, structured and flexible sentences attached to multimedia documents. Further, we want to adapt FSCL for a more general use in knowledge representation and retrieval. Our ideas in this direction are presented in the next sections of this paper.

5 Proposed extensions

We want to indicate several areas of possible changes and extensions to FSCL: changes to its categories and grammar forms; extensions to include ontologies; conversion of FSCL sentences to conceptual graphs to facilitate inferencing; and the introduction of freeform querying.

BEST COPY AVAILABLE

5.1 Changes to categories and grammar of FSCL

As described in section 3, the FSCL categories and grammar have been designed to formulate sentences of the forms 'subject - verb - object' and 'concept - object' in the context of studies of behaviour. To simplify the construction of the vocabulary, adjectives and adverbs have been combined in the FSCL category 'Descriptor' [9]. Adhering to the general FSCL principle of having a formal grammar on fixed, defined categories we are currently investigating a number of changes to FSCL to adapt it to a more general use in knowledge representation. The exact format of the changes has to be determined through applying FSCL in a range of web-based educational systems. Our current thinking centres around the following topics:

- We are investigating changes to the FSCL categories. Merging the categories Person/Thing and Concepts to a more general category, Noun, would address the potential conflict between abstract and concrete terms (see the discussion about the abstract term 'students' and the specific individuals in section 5.3). The category 'Descriptor' could be split up into separate categories of 'Adjectives' and 'Adverbs'. The grammar of FSCL had to change accordingly to accommodate the different roles of adjectives and adverbs within a sentence. The advantage over the current approach in FSCL would be that with this change adverbs could be positioned correctly as in natural language English sentences.

- In natural language, words occur in different grammatical forms in different roles in a sentence ('the instructor *starts* the motor'; 'the motor is *started*'). The current FSCL has a strict separation between its categories. While a word can be defined in its derivations in multiple categories (Activity: starts; Descriptor: started), it is not possible to create a semantic link between the different word forms. We are looking at introducing such a link together with a meta-level grammar to be able to detect semantic equivalence between sentences with word derivatives in different parts of speech.

- The grammar of FSCL could be extended to recognise a wider range of sentence structures. Clausal variations like imperatives ('Start the motor!') or questions ('Is the motor running?') can be introduced. Conditional sentences of the form 'if C then S' would support inference as outlined in more detail in the following section. A wider range of sentence structures recognised correctly by FSCL would increase the potential for knowledge retrieval and inference.

5.2 Extension to use ontologies

FSCL uses hierarchies to define the words of the vocabulary. These hierarchies are defined within the FSCL categories. They are used to group related words and to allow for a retrieval of information on different levels of granularity. These hierarchies, as they are currently used in FSCL, can be seen as simple forms of ontologies. While a number of issues have to be addressed to base FSCL on more substantial ontologies, none of these seems to pose a real problem.

- Users of FSCL define the vocabulary they need for their particular domain. The experience, so far, as reported in [9], show that users define their vocabulary as multiple hierarchies within each FSCL category. These hierarchies could be joined under the FSCL category name to build one ontology within each FSCL category.

- An ontology typically moves from the abstract to the concrete, from concepts to instances. The vocabulary in FSCL is organised in the same way. In a study on 'learning to read', e.g., individual students' names were grouped under the term 'students', individual teachers' names under the term 'teachers' [9]. A term like 'students' contains two components: it has an abstract component in describing a group of the population in general with the property of 'attending school to learn'; it has a concrete component in grouping together specific, named individuals. In the current uses of FSCL this distinction has not caused any problems.

- Not all FSCL categories contain vocabulary which necessarily should be structured as ontologies. While it can be of advantage to organise the vocabulary in the FSCL categories 'Conjunction' and 'Preposition' in hierarchies these words will not build ontologies as they not define 'categories of the world'. Yet the coexistence of ontologies and hierarchies in the vocabulary of FSCL should not create a difficulty.

5.3 Conceptual graphs and inference

FSCL is an easy to understand and effective scheme for an author to create their own vocabulary and use it together with the grammar for describing the contents of a multimedia document such as a video. Currently, knowledge retrieval is performed using the complementary query language FSQL. FSQL addresses the grammatical structure of FSCL sentences, takes advantage of the hierarchy information built into the vocabulary, and offers Boolean, time and sequence query options. However, there is no deductive feature in this scheme which would allow us to be able to infer facts or relations that are not explicitly stated. For example, given the statements:

If anyone starts the motor then the motor is running
The instructor starts the motor

which describes the situation in a training video then we may wish to be able to answer the question:

Is the motor running?

To be able to function at this level, we need the power of a first order logic system. Conceptual Graphs, CG, [20] give us this power.

Our proposal is that the user should describe their domain in terms of FSCL. The statements in this language can then be automatically translated into a CG format. This process is quite straight forward since FSCL is unambiguous, allowing many of the problems of natural language translation to be circumvented.

When a query is made, or some information needs to be located within the document segment then an initial attempt can be made to do this by using FSQL. If this fails then the deductive power of the CG representation is invoked. Standard theorem proving techniques within CG would enable us to check the veracity of a statement. As a bonus, we would get a step-by-step justification of the result proved, similar to the explanation given in expert systems.

5.4 Freeform Querying

Based on a limited yet flexible vocabulary and on a limited grammar, as offered by FSCL and FSQL, a query system can be developed which allows the user to pose questions to the educational system. As the structure and the vocabulary of these questions is known, the educational system can 'understand' these questions. Questions can be mapped against a repository of previously asked questions. If a semantically equivalent question is stored, the corresponding answer is retrieved and presented to the user. If a semantically close question is stored, this previously asked question can be used to facilitate the answering of the new question. As questions and knowledge representation are constructed by the same underlying mechanisms a mapping from question to knowledge representation is possible. This can be used to assist the answering of questions based on the knowledge descriptions and to find the appropriate segments of the multimedia teaching material.

The approach presented here does not attempt to answer any natural language question but a restricted set. The vocabulary is restricted to allow the construction of meaningful questions in a particular domain. The grammar is restricted to allow the construction and comparison of meaningful questions based on the vocabulary. The grammar is generic as it is based on categories which are used to organise the vocabulary across domains. The restriction of vocabulary and grammar distinguishes this approach from the AskJeeves [1] search mechanism. The existence of a grammar distinguishes this approach from keyword based search mechanisms as used in library systems or by internet search machines.

The general idea is to provide the user with specific answers to questions. These answers are retrieved from a body of stored answers only if semantic equivalence can be guaranteed. If semantic closeness is detected the relevant questions with their answers are given to a human operator who then decides on the suitability of the match.

6 Conclusions

In this paper we have considered the need for a knowledge representation mechanism for computer-based educational systems. We have first indicated a number of commonly used mechanisms and have then discussed the Flexible Structured Coding Language, FSCL. We have suggested that FSCL provides an effective mechanism for knowledge representation and subsequent knowledge retrieval, based on the nature of FSCL as a natural language-like description language which allows for flexible, rich yet structured description of learning concepts. As extensions to FSCL we have suggested the integration of more substantial ontologies, the conversion of FSCL sentences into conceptual graph structures and the introduction of freeform querying.

References

[1] AskJeeves, http://www.aj.com/index.asp (accessed on February 2, 2000).

[2] Davenport, G., "Indexes Are 'Out', Models Are 'In'", IEEE Multimedia, Fall 1996, Vol. 3, No.3, pp.10-15, (1996).

[3] Davis, R., Schrobe, H. and Szolovits, P., "What is knowledge representation?" AI Magazine, 14:1, pp. 17-33, (1993).

[4] Decker, S., Erdmann, M., Fensel, D. and Studer, R., "ONTOBROKER: Ontology Based Acess to Distributed and Semi-structured Information", Proceedings of Eighth Working Conference on Database Semantics, Semantic Issues in Multimedia Systems, R. Meersman, Z. Tari and S. Stevens (Eds.), Kluwer Academic Publishers, Boston, (1999).

[5] GENTLE, www.gentle-wbt.com (accessed on April 10 2000).

[6] Gong, Y., Zhang, H. Chuan, H. C., Sakauchi, M., "An Image Database System with Content Capturing and Fast Image Indexing Abilities", Proceedings of the International Conference on Multimedia Computing and Systems, May 1994, IEEE Computer Society Press, Los Alamitos, California, (1994).

[7] Grossmann, R., "The Existence of the World, An Introduction to Ontology", London: Routledge, (1992).

[8] Hampapur, A., Jain, R., Weymouth, T. E., "Production Model Based Digital Video Segmentation", Journal of Multimedia Tools and Applications, 1, 9-46, Kluwer Academic Publishers, Boston, (1995).

[9] Heinrich, E. A., "Multimedia Information System for the Support of Studies of Behaviour", Doctoral Dissertation, Massey University, Palmerston North, New Zealand, (1999).

[10] Heinrich, E. and Kemp, E., "Description and Retrieval across Multiple Media Formats", 4th International Workshop on Query Processing and Multimedia Issues in Distributed Systems, DEXA'00, IEEE Computer Society Press (2000).

[11] Heinrich, E., Kemp, E. and Patrick, J.D., "A Natural Language Like Description Language", 10th Australasian Conference on Information Systems (ACIS), B. Hope and P. Yoong (Eds.), School of Communications and Information Management, Victoria University of Wellington, Wellington, New Zealand, pp. 375 – 386, (1999).

[12] Heinrich, E., Patrick, J.D., and Kemp, E., "A Multimedia Information System to Support the Analysis of Human Behaviour Recorded on Video", Computers in Psychology, Conference Proceedings, CTI Centre for Psychology, University of York, York, (1998).

[13] Jiang, H., Montesi, D., Elmagarmid, A. K., "Integrated Video and Text for Content-based Access to Video Databases", Journal of Multimedia Tools and Applications, 9, pp. 227-249, Kluwer Academic Publishers, (1999).

[14] Lenat, D.B. and Guha, R.V., "Building Large Knowledge-Based Systems; Representation and Inference in the Cyc Project", Addison-Wesley Publishing Company, Reading, Massachusetts, (1990).

[15] Miller, G.A., "WordNet: a lexical database for English", Communications of the ACM, 38:11, pp. 39-41, (1995).

[16] Noldus Information Technology, "The Observer Video-Pro, Version 4.0 for Windows". Wageningen, The Netherlands, (1998).

[17] Palhang, M., Sowmya, A., "Learning Object Models from Real Images", Proceedings of the First International Conference on Visual Information Systems, Victoria University of Technology, Melbourne, Australia, (1996).

[18] QSR Software, QSR NUDIST 4 Overview, http://www.qsr.com.au/Software/N4/qsrn4.htm (accessed on June 22 1998).

[19] Richards, L., "Using NVivo in Qualitative Research", Melbourne: Qualitative Solutions and Research Pty, (1999).

[20] Sowa, J.F., "Conceptual Structures: Information Processing in Mind and Machine", Addison-Wesley, Reading, Massachusetts, (1984).

[21] Sowa, J.F., Knowledge Representation; "Logical, philosophical, and Computational Foundations", Brooks/Cole, Pacific Grove, CA, (2000).

[22] Topic maps. ISO/IEC JTC1/SC34 - Document Description and Processing Languages. http://www.ornl.gov/sgml/sc34/document/0058.htm, (accessed on February 8 2000).

[23] Zhang, H., Gong, Y., Smoliar, S. W. and Tan, S.Y., "Automatic Parsing of News Video", Proceedings of the International Conference on Multimedia Computing Systems, May 1994, IEEE Computer Society Press, Los Alamitos, California, (1994).

Proposal of an XML-based Knowledge Sharing and Management System Supporting Research Activities

Kyoko Umeda*, Takami Yasuda and Shigeki Yokoi***
**Graduate School of Human Informatics Nagoya University,*
***School of Informatics and Sciences Nagoya University*
Furo-cho Chikusa-ku Nagoya-City 4648601 Japan
umeda@info.human.nagoya-u.ac.jp

The proposed system is primarily focused on research activities which create various kinds of knowledge through trial and error. The knowledge is classified into formalized knowledge, such as papers or reports, and un-formalized knowledge, such as suggestions or advices. The former is easily utilized for research activities, because they are accumulated as visible data. However, the latter is not utilized in many cases even if they are informative and useful. Therefore, a web-based management system giving attention to un-formalized knowledge as well as formalized information would be a possible solution. This paper describes the features of the system based on the XML, and shows an example of usage through a trial system. Functions of the system include: (1) collecting un-formalized information related to formalized knowledge, (2) connecting un-formalized knowledge with formalized knowledge, and (3) creating feedback information while using the system. The system creates a repository in a lab, a collaborative space for research activities, and a set of new document and knowledge.

Keywords: **Research Activities, Knowledge Sharing and Management System, Formalized/Un-formalized Knowledge, XML**

1 Introduction

Researches on system environments that share knowledge on the Web have increased because of the needs for accumulating and utilizing knowledge [3][8]. Specially aiming learning activities, the Covis [1], for example, visualizes processes of collaboration between users, and memorizes the processes through the Covis Collaboratory Notebook. Another example is the CSILE [4][9] with networked computer environment particularly designed to support progressive discourse. In CSILE, students write text or graphic notes to convey their explanations. Similarly, the KIE [6] have collaborative environments that make network discussion possible by using the interface called Netbook. Users of the Shrlok [2] also have shared knowledge environments. They can discuss their opinions in an opened condition and make hypertext links between relevant knowledge. Thus, users of these four systems can exchange their own opinions and argue their individual ways of thinking, based on ideas and questions stored in the Database (DB) system [7]. Therefore, in these four systems, students can be subjective while having clear objectives. Teachers can also help students solve problems, and students can collectively work on problems.

The process of advanced researches, on the other hand, is not the same as that of education because researches might not always have definite objectives. In many cases, new things can be discovered from one trivial thought, and researchers enlighten and encourage each other. Individual studies can be more important in a condition where there is no instructive person who clearly knows and ultimate goals. Although research activities have a different characteristic from education activities that have clear goals, few studies aiming research activities have been discussed.

This paper proposes an XML-based knowledge sharing and management system. It focuses on an accumulative style of knowledge management for supporting research activities, rather than for learning.

The activities in a laboratory produce various kinds of knowledge by repeating trial and error. That knowledge is classified into formalized knowledge, such as papers or reports, and un-formalized knowledge, such as suggestions or advices. The former is accumu lated as visible data in the form of paper material or digital data. On the other hand, the latter is only spoken and is not represented in the real material. Therefore most of that information is not recorded. However, it is important to accumulate and share the un-formalized knowledge because live suggestions or advices are often very useful to promoting research activity. Their accumulation is useful for participants to remember knowledge and also for peer that cannot attend the discussion process.

Thus, we focus on this un-formalized knowledge. By making the un-formalized knowledge active as memorandums and by connecting them with meta-data of formalized knowledge, the proposing system creates a new set of knowledge documents, Knowledge DB. Proposing system allows users to produce feedback information while using it. The system by using the XML could effectively help research activities. Finally we provide some considerations on the prototype system.

2 The outline and features of knowledge sharing and management system

Chapter 2 summarizes the features of the proposal.
The system consists of the following three steps.
(1). It attaches un-formalized knowledge with formalized knowledge, for example paper and reports, as memorandums.
(2). It connects the above information with meta-data of formalized knowledge.
(3). It utilizes connected knowledge and feedback the information.

If more than two documents share the same information, they are connected through a memorandums. That is to say, the memorandum connects clearly the original documents existing independently in DB. Such connections are useful for the documents retrieval and research analysis. Further, continuous cycles of connection, searches and analyses can be occurred, which assemble a lot of knowledge and information.

At this time, the trial of this system is focuses on Research DB. However, it is reasonable that fundamental policy is not changed even if the DB is changed, because XML is used for exchanging between applications and our system process only the meta-data.

Three advantages of the system are:

- It provides auxiliary information for user's document retrieval by attaching a memorandum to original documents.
- The original documents are related with each other by the connection with the memorandums, and it creates a new document set.
- It supports continuous research activities for users to analyze sets of information and knowledge.

3 Adoption of XML technology

Chapter 3 discusses advantages of the XML, which is one important characteristic in the system.

We adopted the XML, a standard language for information exchange, for two reasons. The first was the need to do knowledge management on the Web because the sharing space accumulated knowledge can be accessed anytime and anywhere. The second was the need to consider the connection with another advanced DB, such as CG and 3D data. Thus, the system would be more flexible because of the XML.

Effectively, the XML is used in two aspects. One is as a way for exchange between DB and systems. The other is for the preservation of information, including the XLink function [10]. Considering that memo randum and data items can change in near future, XML has several advantages: It can set flexible data lists, and express arbitrary number of elements in a tree structure [5].

4 The system configuration

Chapter 4 shows the configuration of the system.

The system consists of three main parts: (1) Sets of Knowledge-Memos, (2) Knowledge processing system, and (3) Interface for knowledge sharing on the Web. (Fig.1). The role of the part (1) is collection and accumulation of knowledge. Part (2) connects two kinds of knowledge. Part (3) relates to the interface for users. The followings sections present their details, respectively.

4.1 Set of Knowledge-Memos: Collection, accumulation of the memorandums

The system needs to collect un-formalized knowledge, such as advices or suggestions from teachers and researchers, even though they are not in any form. Thus, the style of memo randums to formalized knowledge, like papers, are adapted. This chapter presents the concept of "Knowledge-Memo".

4.1.1 The proposition of the Knowledge-Memo concept

The system adopts concept of memorandum called "Knowledge-Memo", in order to collect un-formalized information. We classify Knowledge-Memos into two types to be attached to the original documents in accordance with their natures. In this way, layers of un-formalized knowledge can be created.

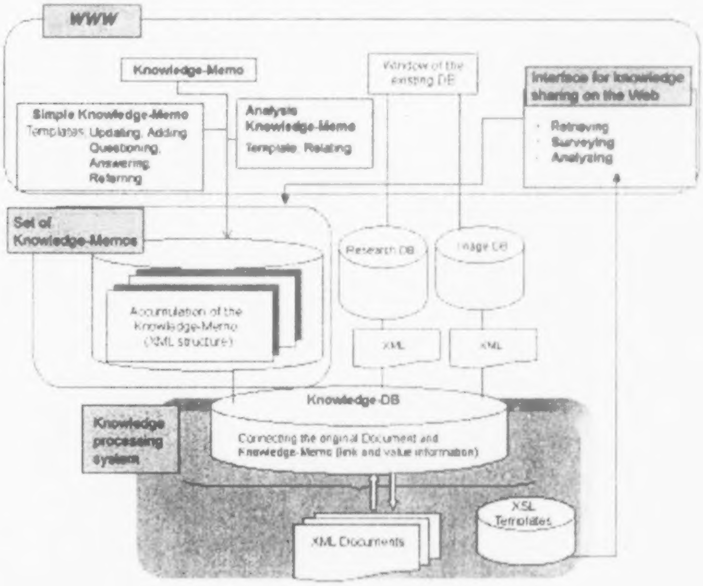

WWW
Knowledge-Memo
Window of the existing DB
Interface for knowledge sharing on the Web
Simple Knowledge-Memo
Templates: Updating, Adding, Questioning, Answering, Referring
Analysis Knowledge-Memo
Template: Relating
Retrieving
Surveying
Analyzing
Set of Knowledge-Memos
Research DB
Image DB
Accumulation of the Knowledge-Memo (XML structure)
XML
XML
Knowledge-DB
Knowledge processing system
Connecting the original Document and Knowledge-Memo (link and value information)
XSL Templates
XML Documents

Fig.1. System overview

Simple Knowledge-Memo: specific information which users want to attach. For example, "This paper is an updated version of named B paper." This type of memo randum can be registered at the same time original paper is entered in the DB.

Analysis Knowledge-Memo: constructed and connected information that is based on researchers' analyses. This type of memo randum can be a Simple Knowledge-Memo because it can be re-analyzed. Users would register Analysis Knowledge-Memo as research results of documents and memorandums.

According to making of the Knowledge-Memo, new sets of documents are created. One objective of proposed system is to change from fragmentary and separated information to collected new knowledge, due to the analyses of researchers in a common created space.

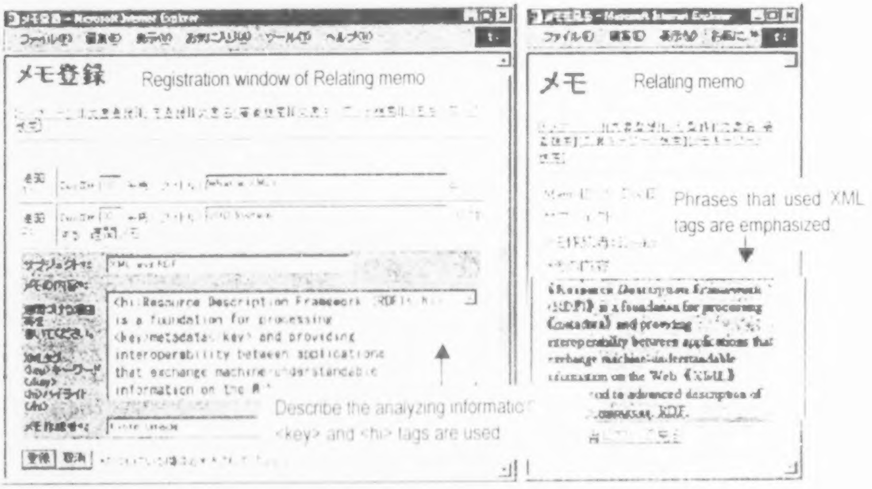
4.1.2 Collection and accumulation of Knowledge-Memo

BEST COPY AVAILABLE

The following templates make inputting memorandums simple.
Information inputted in prepared templates is stored on the Web as Knowledge-Memos through XML structure. Types of the Simple Knowledge-Memo are updating, adding, questioning, answering and referring. Analysis Knowledge-Memo includes relating memorandums.

Usage of these templates is as follows.
Updating templates: describing information and explaining reasons for renewal, which create relationships between before and after renewal.
Adding templates: adding information, such as advices and references to original documents.
Questioning templates: asking questions to documents. When inputting Questioning templates, e-mails would be simultaneously sent to a person who created the original documents.
Answering templates: answering to questions. Automatically sent to the person who wrote questions.
Referring templates: referring to external documents and create new relationships with sites on the Web.
Relating template: describing relationships between documents which are based on analysis of documents and Knowledge-Memos. More than two documents and memos can have relationships.

Several tags of the XML are also used• •<key> for keywords, • <hi> for highlights, • <br /> for starting new lines. In an experimental usage of the system, users were free to use these tags without any restriction and enforcement. If tags were used, words would be shown in only emphasized style on the screen (Fig.2). However, the system would better more reflect users' intentions if the use of new tags were available and inventive Extensible Stylesheet Language (XSL) was developed.
As previous discussion shows, the system has an advantage of creating sets of documents, which reflects users' intentions.


メモ登録
Registration window of Relating memo
メモ
Relating memo
Phrases that used XML tags are emphasized
Describe the analyzing informatio
<key> and <hi> tags are used

<hi>Resource Description Framework (RDF)</hi> is a foundation for processing <key>metadata</key> and providing interoperability between applications that exchange machine-understandable information on the W

Fig.2 Input screen of "Relating memo" used to input XML tags (Left)
The "Relating memo" including enhanced expressions created through XSL (Right)

4.2 Knowledge processing system: Connecting the original document and Knowledge-Memo

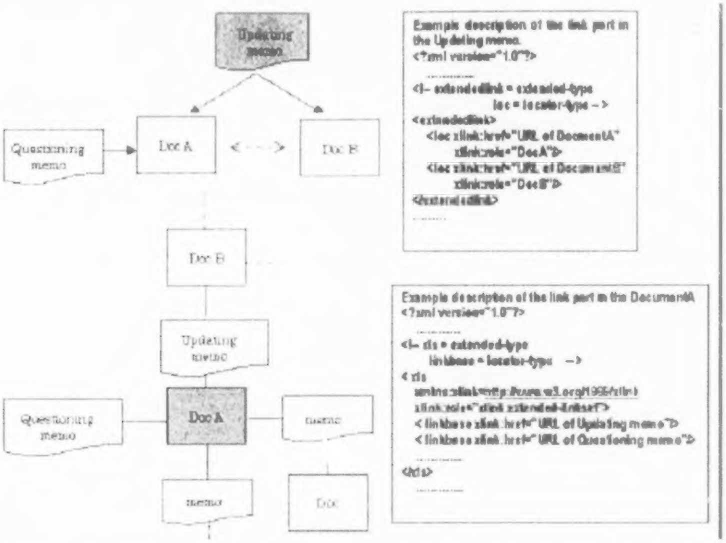
After collecting un-formalized information, the system connects it with formalized information. Such connection creates a Knowledge repository.

The process of connection is as follows. First of all, this system picks up necessary meta-data from Research DB and stores it in a XML structure. Such information is connected to the Knowledge-Memo which is also in a XML structure. Thus, a Knowledge repository is created. The system employs XLink function to connect un-formalized information with documents. Because of XLink potential, it is possible to make multidirectional links among original documents from a remote resource, that is, from a Knowledge-Memo related to original documents. Moreover, the system also creates lists of linkage

BEST COPY AVAILABLE

information about existing Knowledge-Memos related to one original document. That is, from one individual document all its existing connections are easily obtained (Fig.3). Unfortunately, the experimental utilization of the system in this paper uses Internet Explorer5 which still does not support all these XLink functions. That is why the system utilizes link functions of HTML, reflecting the structure of the XLink. If the XLink was supported, it would be easily possible to make relationships between documents through the above simple structure. The fact that these connections are automatically created by users' simple operation constitutes an advantage of the system.

The Knowledge DB pulls out necessary information, and displays on a Web interface. The system uses XSL templates to arrange and display requested information.



```
Example description of the link part in
the Updating memo.
<?xml version="1.0"?>
  ..........
<!-- extendedlink = extended-type
              loc = locator-type -->
<extendedlink>
  <loc xlink:href="URL of DocumentA"
       xlink:role="DocA"/>
  <loc xlink:href="URL of DocumentB"
       xlink:role="DocB"/>
</extendedlink>
.........
```

```
Example description of the link part in the DocumentA
<?xml version="1.0"?>
  ..........
<!-- xls = extended-type
    linkbase = locator-type   -->
< xls
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xlink:role="xlink:extended-linkset">
  < linkbase xlink:href="URL of Updating memo"/>
  < linkbase xlink:href="URL of Questioning memo"/>
  ..........
</xls>
  ..........
```

Fig.3 Description examples of relation between documents and Knowledge-Memo based on XLink. The memorandum associates remote documents through extended link (above). The external linkset centralizes the link information (below).

4.3 Interface for knowledge sharing on the Web

Peers use a trial system on the Web as a part of research activities. In order to make a user-friendly interface, we studied the flow of research activities. As the result, three processes, such as retrieving, surveying and analyzing information, are prepared for their research activities.

First, two retrievals are available, which include searching documents and Knowledge-Memos. Document search is a method which is often used, and it searches a document from a title or keyword. If an Updating memo is shown as a result, and there are some corrections on the documents including updated document. In another word, Updating memo provides help of the retrieval. Moreover, a renewal reason has the possibility to become a reference when a peer writes a paper. Retrieving from Knowledge-Memos may be useful for getting information toward vague ideas. I can be more efficient than previous ways, because researched results are sets of documents and memorandums. Further, due to the XSL, it is possible to sort by dates and to filter by types of memorandums.

In a Surveying process, connection between documents and memorandums is visualized, when traversing search results. For example, even if users think that there is no relationship between documents, there might have some kind of relationship after following links. Such new researches can help proceeding researches.

BEST COPY AVAILABLE

With respect to analyzing information, a new finding, resulted from surveying information, can be used for making analyzing memo in a combination with related and added memo randums. These processes can be continued by adding new information and findings that stimulate utilization. On the Web, a common space, such utilization can increases effective research activities.

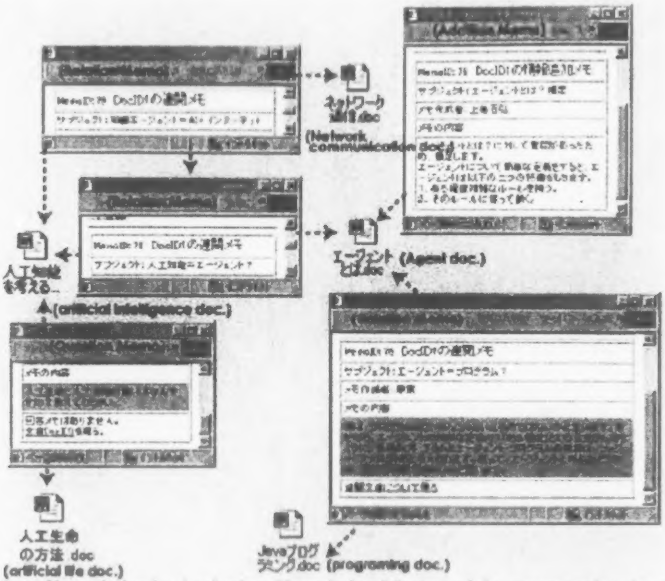

ネットワーク通信.doc
(Network communication doc.)
エージェントとは.doc (Agent doc.)
人工知能を考える...
(artificial intelligence doc.)
人工生命の方法 doc
(artificial life doc.)
Javaプログラミング.doc (programing doc.)

Fig.4 Set of knowledge by Analyzing; Knowledge-Memo and documents related to it. Documents and Memos are gathered around the "Agent document".

5 Prototype evaluation

Usage of the system and evaluation of the prototype are discussed and reviewed in this section.

5.1 Usage of the system

In order to study further, followings show a way of system utilization, based on discussions and reports in a research group which studies agent technologies in a laboratory. Suppose that there are three members, named A, B and C, in the group.

(1) "A" makes and reads a report, "About Agent" in a seminar. After the seminar he registers the report in Research DB. At the same time, conclusion of discussion, advice, etc. are also registered as Knowledge-Memos.

(2) "B" who was absent for discussion reads the report. Then "B" asks, "What does autonomy mean?" in a Questioning memo. Such question is registered in memo randums of questions related to reports, and at the same time, "A" will get the e-mail.

(3) "A" answers the question from "B" in Answering memos, which is registered in Answering memos, and e-mail is sent to "B".

(4) "C" tries to do a programming of an agent by using Java. He finds a report of "About Agent" written by "A" through a keyword search, "agent". "C" completes his report, referring A's report. . He makes a Relating memo, for example, describing which part of the agent report is quoted and how it is useful for him.

After repeating these memorandums registrations, it is possible to analyze information as shown Fig.4. Members of agent seminar could gain the following effects at this time.

5.2 Test results

Seven students in a lab used a practical sample test of the system, and answered questionnaires. Table.1 shows the results.

	Yes	No
(1). Helpfulness of the Knowledge-Memo.	93%	7%
(2). Easiness of inputting the Knowledge-Memo.	58%	42%
(3). Acceptance of sharing ideas written in the memo with other users.	93%	7%
(4). Satisfaction of inserting XML tags for emphasizing and changing colors.	67%	33%
(5). Usefulness of the system. (i.e. connecting the Knowledge-Memo with documents, resulting in a set of new documents.)	71%	29%

Table1. Results of questionnaires.

The overall evaluation of the system was positive. In terms of the question (1), students used the Knowledge-Memo for connecting to related documents and getting information of their documents. There were several responses in question (2), which demanded for the improvement of the interface when inputting the memorandums. Some students suggested a possibility of creating more successful system if incorporating with other laboratory members. In the question (3), most students were positive for informing and sharing ideas through memorandum with other users, since they can identify their ideas and get some comments. As for question (4), some students complained the new tag system that requires additional input. However, other students recognize the advantages of the system that can emphasize the keyword and change colors as far as the tags were not so complicated. Finally, most students recognize the structure of relationships centered on the document is useful for research activities.

5.3 Discussion

Test results lead to three fundamental findings.

(1). The system is useful for using and searching documents because it is possible to use information of Knowledge-Memo as well as abstracts.

(2). The system is convenient, since it enables users to make relationships with other preserved documents, to create new sets of documents, and to traverse from memorandums.

(3). The system is effective for informing and sharing opinions with peers because it enables to identify their ideas, to get some comments, and to record the process of studies.

From these results, it is possible to conclude that this management system effectively supports research activities, which collects and accumulates peer's knowledge and promotes collaborative and shared utilization.

Furthermore, we need to evaluate more effects for future research, such as;

- Is there any possibility in this system to give linkage of documents that seem to have no relation with each other?
- Is there any possibility that the results of using this system, such as creation of new document sets and analysis of memorandums, can give deeper understanding and new definition to users?

Additionally, this system should be improved in terms of the following three points.

(1). Revision of interfaces, including the interface for inputting the memo and the interface for classifying documents by theme.

(2). Addition of the level of importance to Knowledge-Memos for arrangement and classification, in order to promote re-use of knowledge.

(3). Exploration and employment of XLink potential. (Current browsers, such as IE or Netscape communicator, do not support XLink functions.)

6 Conclusions

The proposed web-based management system is primarily designed for research activities. Previously, databased and written information, such as papers and reports, were only available for research activities, even though other information, such as ideas and opinions, are also important knowledge. The new management system enables to utilize un-formalized knowledge as well as formalized information.

Positive responses from lab members who used a trial system show that because separated and fragmentary information are collected through Knowledge-Memos, effective and efficient research activities would be feasible. A lot of information and ideas toward papers are collected by members as databases, which creates sets of documents. Researchers can collaborate with other researchers through the system.

From the technical standpoint, the system utilizes the XML in two parts of exchange and preservation. Users' intentions on the WWW can be more reflected by the XML.

For the future usage, since only meta-data is managed in a XML, the utilization of documents as well as digital data is feasible. Further, the system can connect knowledge more easily, since XLink functions will be realized soon. Important advantages of the system include creation of relationships, and searches of information and knowledge. Improvement of the interface and the classification memorandums will be necessary for the long term.

References

[1] D.K. O'Neill, L.M. Gomez, "The Collaboratory Notebook: a Networked Knowledge-Building Environment for Project Learning", Proc. of Ed-Media 94, pp.416-423 (1994)

[2] H. Ogata, K. Imai, K. Matsuura and Y. Yano, "Knowledge Awareness Map fro Open-ended and Collaborative Learning on World Wide Web", Proc. of ICCE99, pp319-326 (1999)

[3] I. Nonaka, "The Knowledge-Creating Company", Harvard Business Review, November-December1991, pp96-104 (1991)

[4] J. Oshima, "Students' Construction of Scientific Explanations in a Collaborative Hyper-Media Learning Environment", Proc. of CSCL97, pp187-197 (1997)

[5] N. Pitts-Moults, C. Kirk, "XML Black Book", The Coriolis Group. Inc. (1999)

[6] P. Bell, E.A. Davis, M.C. Linn, "The Knowledge Integration Environment: Theory and Design", Proc. of CSCL95 (1995)

[7] S. Puntambekar, "An integrated approach to individual and collaborative learning in a web-based learning environment", Proc. of CSCL 99, pp458-465 (1999)

[8] T.H. Davenport, "Integrating Knowledge Management into the Organization", Diamond Harvard Business, August-September 1999 pp26-36 (1999)

[9] Knowledge Forum 2.0, http://www.motion.com/lim/kf/KF1.html (accessed on August 1, 2000)

[10] XML Linking Language (XLink) Version 1.0, http://www.w3.org/TR/2000/CR-xlink-20000703/ (accessed on August 1, 2000)

A Real-time Handwriting Communication System for Distance Education[+]

Hsin-Chu Chen* Jho-Ju Tu and Jianping Zhang*****
Department of Computer and Information Sciences, Clark Atlanta University , 223 J.P.Brawley Dr., SW, Atlanta, GA 30314. hchen@star.cau.edu.
** *Instructional Technology Center and School of Education, Georgia Southwestern State University, 800 Wheatley St., Americus, GA 31709. tu@canes.gsw.edu*
*** *Department of Computer Science, Utah State University, Logan, Utah 84322. jianping@zhang.cs.usu.edu*

1 Introduction

In this paper we present the design and implementation of a handwriting communication system for real-time graphical information exchange. This system provides an environment for a user to write and erase messages on a computer screen using a light pen or a mouse and to transmit the handwriting message to another user on the Internet in real time. The communication techniques employed for the system include the basic client-server model and peer-to-peer model. The client-server model is mainly for sending handwriting information using the world wide web. The peer-to-peer model, however, is aimed at real-time communications between two end users on the network to conduct instant dialogues. The system is implemented using Java . It can be integrated into many different applications such as collaborative learning, on-line discussions, and distance education.

2 Communication Models

A handwriting communication system may be implemented using a client-server or a peer-to-peer model. Each model has its cons and pros. The client-server model, in which the client sends requests to the server and the server responds to the request [3], works well in situations when immediate responses are not necessary. For a real-time instant dialogue or information exchange, however, the client-server model alone is somewhat restrictive due to its limited degree of interactivity. To achieve a full degree of interactivity for this type of applications, a peer-to-peer model that allows for full duplex real-time communications is more appropriate, since the two end users of the system may send and receive information at the same time, roughly speaking.

In addition to communication models, we must also take into account the nature of the communication protocols and decide which to use. Information exchange carried over the Internet normally requires support from either TCP or UDP, which are the two protocols operated at the transport layer in the TCP/IP protocol suite [1]. When TCP is employed, the information is sent as a data stream, similar to a telephone conversation. Since TCP requires a connection setup prior to transferring data, it incurs an initial time delay. UDP, on the other hand, does not require such a connection setup. However, the delivery of datagram packets, which are independent data units sent individually from the source to the destination, is not guaranteed. Datagram packets may arrive out of order too. For textual information, UDP may not be a bad choice because the user normally can tolerate, to certain degree, occasional loss of packets or misplaced textual data. In our handwriting system, the handwritten information is represented as numerical data which are sensitive to the loss of any single bit of information, therefore, TCP is our natural choice.

[+] This work was supported in part by the Army Research Laboratory under grant DAAL01-98-2-D065.

BEST COPY AVAILABLE

3 System Design

Our design philosophy for the development of the handwriting portion centers around the following principles: interactivity, functionality, reliability, user-friendliness, and portability. A handwriting communication system must possess two important capabilities. The first is the ability to support the writing by a light pen, or a mouse if such a pen is unavailable, and the display of the handwritten data. The second capability should allow users to transmit and receive handwritten data from the network. To achieve these goals, a friendly graphical user interface, which requires the use of abstract windowing toolkit and event handling, is a necessity. In order to make the system a useful communication tool on the Internet, it must support both textual and graphical information exchanges. The system must also provide functions for users to overwrite or modify handwritten message received over the network. In addition, the programming language used for implementation must be platform independent so that the system can be easily ported to other machines with different operating systems.

4 Description of the System and its Applications to Distance Education

As mentioned earlier, we use Java [2, 4] as the programming language and TCP as the transport protocol for transferring handwritten data in our current client-server communication system. A graphical user interface consisting of buttons, radio check boxes, and a handwriting area, as well as the operations associated with the interface have been developed using the abstract windowing toolkit. All main tasks of the system are invoked from within the event handling functions. Our system currently allows users on the Internet to exercise handwriting from within a web page that contains the client code and send the information to the server that accepts the handwritten data. It can also be used to enhance online presentations over the Internet. This is due to the fact that the system allows users to perform handwriting directly on the specified writing area in a web page. By putting the presentation material inside the handwriting area, it is possible to add notes, make corrections, highlight important subjects on the spot during the course of the presentation.

The handwriting communication system has many applications in distance education and on-line collaborative learning. It can be used by an instructor to deliver on-line lectures via the web; the instructor may use one part of the screen to present prepared presentations and another part to highlight the important points of his/her presentation using a light pen. It can be used by fellow students in different locations to solve problems collaboratively and work on team projects. In addition, the instructor and students can use it to conduct on-line class discussions and answer student' s questions by employing the communication capability.

5 Conclusions

In this article, we have presented the basic approaches, design considerations, and implementation of a real-time handwriting communication system on the Internet as well as its applications to on-line education. Our design philosophy centers around functionality, interactivity, portability, and user friendliness.

References

1. Andrew S. Tanenbaum, "Computer Networks", NJ: Prentice Hall, 3rd Edition , (1996).
2. David Flaganan, "Java in a Nutshell, Sebastopol", CA: O' Reilly & Associates, (1996).
3. David Gunter, et al., "Client/Server Programming with RPC and DCE", IN: Que, (1995).
4. Patrick Naughton and Herbert Schildt, "JAVA: The Complete Reference", CA: McGraw-Hill, (1997).

The Application of Uncertainty Reasoning for an Intelligent Tutoring System

Hown-Wen Chen*, Teran Wu and Dong-Yih Bau
Institute of Information Management, Da-Yeh University
112, Shan-Jeau Rd., Da-Tsuen, Chang-Hwa, Taiwan
Email :hwchen@aries.dyu.edu.tw

The activity of test and evaluation is an important part of Computer-Assisted Instruction systems. In most systems, 「absolutely learned」 and 「absolutely unfamiliar」are often used to represent the status of a student in learning a novice concept. However, for each target concept, there are usually more than one related sub-concepts with different degrees of importance. Thus, it is quite difficult to instruct each individual student effectively according to his learning status in those conventional systems. A hybrid technology of fuzzy theory and uncertainty reasoning are thus used in the research. The proposed intelligent tutoring system was designed to illustrate: 1.automatically tracking and analyzing the current learning status of a pupil, especially detecting the formation of learning barriers or misconceptions; 2. autonomously leading pupils to visit assisted learning path and thus proposing tutorials to make the learning of students more effectively. 3.linguistically explaining the implicit behavior of a pupil during the whole learning process. In addition, the mathematical course of teaching Pythagoream Theorem was used as the content of our test-bed. A simulation by hand and positive feedbacks from teachers of junior high schools illustrate the reasonableness and applicability of the proposed tutoring system.

Keywords: **Pythagoream Theorem, Fuzzy Logic, Uncertainty Reasoning, Intelligent Tutoring System**

1 Introduction

Researches about Intelligent Computer Aided Instruction (ICAI) have incrementally grown since 1970, for example, standard intelligent tutoring systems [1], or participants in virtual environments [2], or a virtual instructor in a training environment [3]. However, as known, the effectiveness of education would depend on the local culture. But, there are few intelligent tutoring systems focusing on Taiwanese students have been reported. CORAL [4] was designed as an interface system, without any artificial intelligence module of teachers' expertise, to provide a long-distance collative learning environment of virtual learning. As discussed in lots of tutoring systems, the most challenging issue is how to evaluate and diagnose the learning of students. Tests are a typical and popular method of evaluation. Taking the GRE as an example, people have taken the test through computers since 1992. The IBM co. and Arthur Anderson Co. have begun to work on the development of a computerized testing system. Such systems, which change the form of tests from conventional paper-to-pencil to on-line, are proliferating rapidly. For ICAI, it becomes more popular that the evaluation of pupils' learning should not be simply classified as「absolutely learned」and「absolutely unfamiliar」. In addition, ways of leading each individual pupil to enjoy an efficient learning experience is also pursued. In the research, we proposed an intelligent tutoring system which can afford the most appropriate tutorials to each pupil according to his learning status and thus can prevent pupils to trap into a misconception too long.

2 The Organization of Tutorials and Maintaining Principles

Before implementing our tutoring system, some special issues and adopted techniques must be introduced.

BEST COPY AVAILABLE

Those topics include the organization of tutorials, a way of representing pupil's learning status, and the detection of any formed misconception.

2.1 The Construction of a Hierarchical Concept Tree

In general, tutorials would be organized as a tree hierarchy of curriculum in the order of chapter, section, sub-section, paragraph, etc. Since learning a complicate concept must depend on the success of learning all its related sub-concepts, the kind of structure cannot be claimed to be suitable for both learners and instructors. That is, too few containment or precedence information about curriculum is available. Thus, learning concepts and related tutorials are re-arranged as a hierarchical conceptual tree of containment here. According to literatures [5] and interviews with teachers of junior high schools, the concepts related to learning Pythagoream Theorem for native pupils can be analyzed and constructed as Figure 1. In the tree, the learning of any parent conceptual node must follow after at least one of its children nodes.

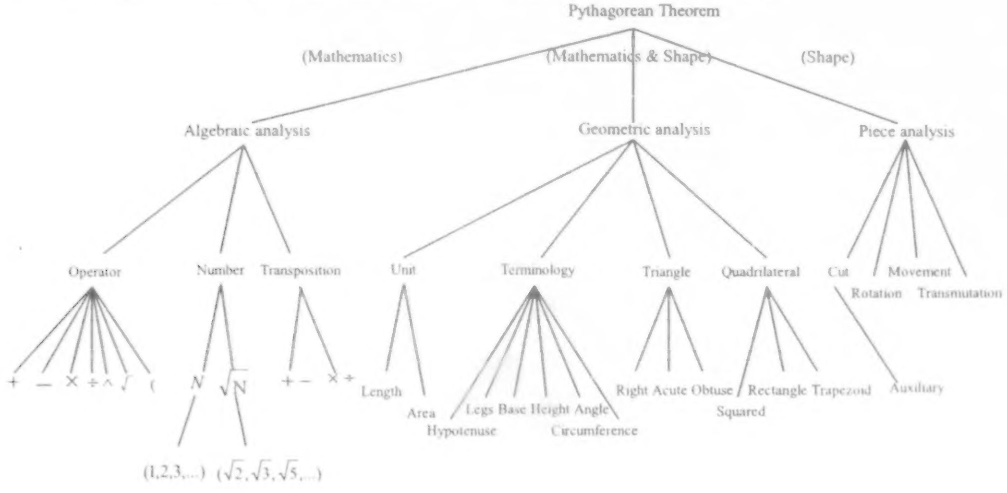

Pythagorean Theorem
(Mathematics)
(Mathematics & Shape)
(Shape)
Algebraic analysis
Geometric analysis
Piece analysis
Operator
Number
Transposition
Unit
Terminology
Triangle
Quadrilateral
Cut
Movement
Rotation
Transmutation
+ — × ÷ ^ √ (
N √N
(1,2,3,...) (√2, √3, √5,...)
+ − × ÷
Length
Area
Legs Base Height Angle
Hypotenuse
Circumference
Right Acute Obtuse
Rectangle Trapezoid
Squared
Auxiliary

Figure 1. A hierarchical concept tree of Pythagorean Theorem

2.2 The Setting of Node Weights within the Hierarchical Concept Tree

To express the corresponding degree of importance, an integer is assigned to each testing question related to individual concept [5]. However, it is still a heavy burden even for an expert to quantitatively assess the extent. Besides, the estimated grade of importance is too subjective in general. In our system, the influence of each node on learning its parent node is defined through fuzzy theory as follows:

Step 1: Some teachers in junior high schools are asked to evaluate the relevance of nodes related to their parent node in the hierarchical concept tree.

Step 2: Fuzzy theory is included to quantify teachers' opinions in the designed questionnaire obtained in step 1. Five possible values for linguistic variables are used. Note magnitudes 0.0 and 1.0 are not adopted in the memberships because of product operations and symmetry.

Step 3: Murray's or Ishihawa's Max-Min method is used to fuzzily integrate those multiple expertise. After that, a defuzzification process to evaluate the mass centroid of fuzzy numbers is applied. The weights of nodes within the hierarchical concept tree are thus settled as shown in Figure 1.

2.3 The Maintenance of Belief Parameters

To increasing the expression power of the proposed system above "absolutely known" and "absolutely unfamiliar", a belief parameter m and another updating parameter λ described in Dempster-Shafer Theorem [6] are applied here to assess the familiarity degree of a pupil to a particular learning concept within the hierarchical concept tree. To describe the meaning of the updating parameters λ and θ ($\theta = 1 - \lambda$), two cases must be taken in account:

Case 1: Making a correct answer

BEST COPY AVAILABLE

λ and θ can be used to denote the belief degree of promoting to a higher level and of staying on the same level within the conceptual hierarchy, respectively.

Case 2: Making a wrong answer
λ and θ can be used to denote the belief degree of degrading to a lower level and of staying on the same level within the conceptual hierarchy, respectively.
As to defining the updating rules of the belief parameter m, a general sub-tree structure is considered. In the tree, a node f has three children nodes labeled as a, b, c, and the interconnection links are labeled as W_{af}, W_{bf}, W_{cf}

Case 1: *Making a correct answer in the test for the conceptual node a*
A promotion within the conceptual hierarchy must be activated. The belief parameters of the two relevant nodes a and f are thus modified as

$$\begin{aligned} m_f^t &= (W_{af} \bullet \lambda) + m_f^{t-1} \\ m_a^t &= (W_{af} \bullet \theta) + m_a^{t-1} \end{aligned} \qquad \textbf{(eqn. 1)}$$

m_f^t, m_a^t : the magnitudes of belief after promotion
m_f^{t-1}, m_a^{t-1} : the magnitudes of belief before promotion
W_{af} : the weight of link between nodes **a** and **f**

Case 2: Making a wrong answer in the test for the conceptual node **f**
A degradation within the conceptual hierarchy must be activated. The belief parameters of the four relevant nodes, **f** and its children nodes **a, b, c,** are thus modified as

$$\begin{aligned} m_f^t &= (\theta) + m_f^{t-1} \\ m_a^t &= [\, W_{af} \bullet \lambda \bullet (1 - m_a^{t-1}) \,] + m_a^{t-1} \\ m_b^t &= [\, W_{bf} \bullet \lambda \bullet (1 - m_b^{t-1}) \,] + m_b^{t-1} \\ m_c^t &= [\, W_{cf} \bullet \lambda \bullet (1 - m_c^{t-1}) \,] + m_c^{t-1} \end{aligned} \qquad \textbf{(eqn. 2)}$$

Case 3: If a correct answer is made in the topmost conceptual node, it is impossible to promote anymore. However, the belief of the topmost conceptual node is still updated with eqn. 1.

Case 4: If a wrong answer is made in the lowest conceptual node, it is impossible to degrade and the belief of the node is updated with eqn. 2.

2.4 The Strategy of Instruction

Several principles have been applied in the proposed system:
The instruction and assessment examination would only take place in the conceptual node with the largest belief. However, all assessment tests for its children nodes with weights larger than a pre-chosen threshold must be answered correctly. If the mentioned condition is not satisfied, the focus of instruction and assessment would be transferred to one of its children nodes instead.

According to Dempster-Shapfer Theorem, the procedure of normalization must be applied after each updating of belief.

There is an implicit relationship between the magnitudes of weights and belief parameter λ. To avoid the learning process to be not in progress, according to eqn. 2, the magnitude of belief updating in any child node (a) must be larger than that of parent node (f). Thus,

$$W \bullet \lambda \bullet (1) > 1 - \lambda$$

$$\Rightarrow \lambda > \frac{1}{(1 + w)} \text{ for all possible w}$$

2.5 The Analysis of Learning Traces and Detection of Misconceptions

Two kinds of traversal information would be recorded during the learning process: the weighted correct rate of answering testing questions for each conceptual node, and the traversal path of all visiting nodes.

First, the weighted correct rate can be used to indicate the current comprehension degree of a concept during the learning progresses. As known, the status near to the ending of learning should be emphasized. In other words, a pupil would be regarded as having been familiar with the concept if he can finally pass the corresponding test independent of times of previous failures. To simulate the phenomenon, three kinds of information must be kept: the number of making wrong answers W, the number of making contiguous correct answers after the last wrong answer C, and the total number of answering T. The weighted correct rate is defined as $1-W / [(T-W-C)+W+2*C]$, i.e., $1-W/(T+C)$. The interpretation of the weighted correct rate would be based on fuzzy expression in our system.

Another important issue is the way of detecting the formation of a misconception. A misconception may be caused by some blind spots of learning and thus always makes the learning process trap into a loop. A good diagnosis module of a tutoring system must have such kind of detection capability and could inform the other tutorial guidance module to show some appropriate auxiliary tutorials. If the test of each child node has passed, i.e., the learner has traversed and correctly answer all questions related to the concepts of all children nodes, the conceptual node is marked as P (Passed). If a learner cannot pass the test of a conceptual node and all its children nodes satisfy one of the following two conditions, then the learner is identified as trapping in a misconception corresponding to the conceptual node. The two conditions are <i>the child node has been marked as P; or <ii>the weighted correct rate is absolutely 1 (100%).

3 The Development and Design of Our System

Based on those described ideas, a prototype tutoring system comprising a testing and evaluation module has been developed and demonstrated. Microsoft Visual FoxPro 6.0 is used under the platform of Microsoft Windows 98. There are four modules included in our system shown in Figure 2.

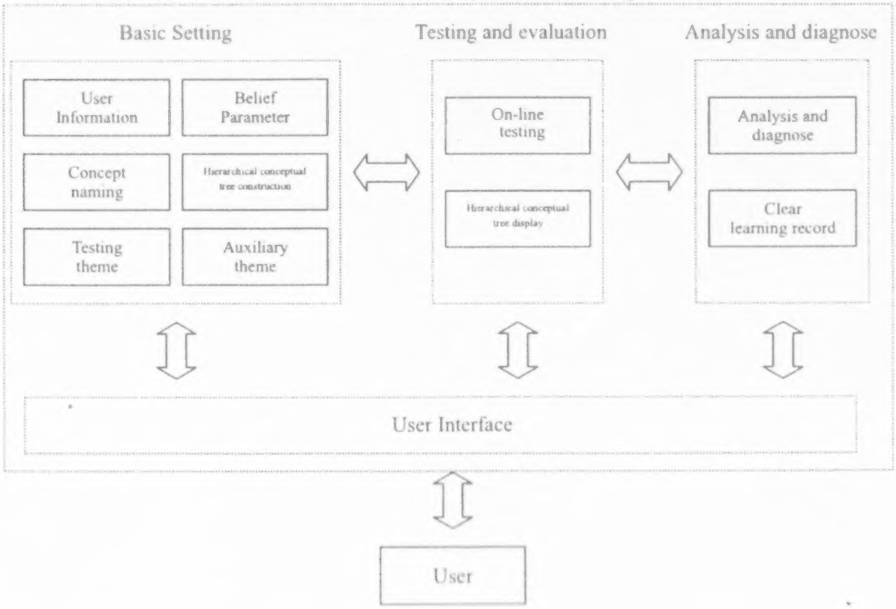

Basic Setting
Testing and evaluation
Analysis and diagnose
User Information
Belief Parameter
Concept naming
Hierarchical conceptual tree construction
Testing theme
Auxiliary theme
On-line testing
Hierarchical conceptual tree display
Analysis and diagnose
Clear learning record
User Interface
User

Figure 2. The architecture of the tutoring system

4 Conclusion and Future Work

In the research, techniques of fuzzy theory and uncertainty reasoning are applied to create a novel tutoring system. As demonstrated, the proposed tutoring system shows an excellent capability to present proper tutorials to guide pupils, precisely evaluates their learning status, and then shows auxiliary teaching materials to prevent pupils from trapping in any formed misconception. Finally, the traversal of learning would be analyzed and interpreted by fuzzy expressions.

Besides, some issues are worthy of deeper investigations through the study:

1. Some adaptive techniques of machine learning, e.g., genetic algorithm and artificial neural networks, should be applied to help instructors to automatically choose or tune parameters used in the tutoring system.
2. More applications about the proposed system should be examined to show its portability.

Acknowledgement

This study is supported by National Science Council of R.O.C. under contract number
NSC88-2520-S-212-003

References

[1] Anderson, J.R,"Cognitive modeling and intelligent tutoring", *Artificial Intelligence*. 42, pp.7-49, (1990).

[2] Pimentel, K. and Teixeira, K, "Virtual Reality: Through thenewlooking glass", Blue Ridge Summit, PA: Windcrest/McGraw-Hill, (1994).

[3] Rickel, J. and Johnson W. L, "Pedagogical agents for immersive training environments", Proc. of the 1st International Conf. on Autonomous agents, New York, (1997)..

[4] Chien Chou and Chuen-Tsai Sun, "The design and implementation of a cooperative distance learning environment: The CORAL experience", Education Technology and Media, 26, pp. 13-21, April, (1997).

[5] C. S. Hsu, H. F. Dou and G. J. Huang, "The application of concept inheritance in learning diagnosis system", Proc. of ICCAI'98, pp. 602-609, (1998).

[6] Shafer, G., "Mathematical Theory of Evidence ", Princeton University Press , Princeton , New Jersey ,(1976).

The Automated Teaching Assistant: Automatic construction of teaching materials from course outlines

Chris Houser*, Shigeki Yokoi and Takami Yasuda****
**Kinjo Gakuin University, Omori 2-1723 Moriyama-ku Nagoya 463-8521 Japan, houser@kinjo-u.ac.jp*
***Nagoya University, Furo-cho Chikusa-ku Nagoya 464-8601 Japan*

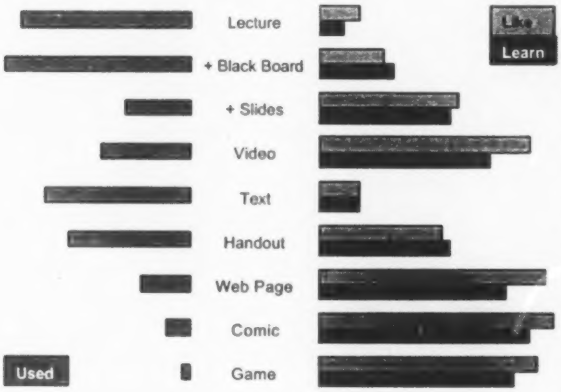
Keywords: **Lecture Preparation, Automation, Design of Teaching Materials.**

1 Educational Media: Yet Another Digital Divide

Instructors can use many media. Traditional teachers lecture while outlining on a black board; they assign readings from texts and printed handouts. Technological teachers lecture (or direct activities) while showing PowerPoint slides; they assign readings of web pages.

Instructors can use many media. Which do they in fact use? Which do students prefer?

We polled 200 students at three Japanese colleges. We asked students how often they saw, how well they liked, and how well they learnt from nine educational media. Responses were similar for all students across different colleges, different grades, and different majors. This graph summarizes their unified opinion.



The left side shows how frequently students saw each media. Common were traditional media (lecture, black board, text, and handout). Modern media were rare.

The upper bars on the right show how well students liked each media. The modern, rarely used were well liked. The lower bars show how well students thought they learnt from each media. Students claimed that well-liked media also taught the best.

In interviews, teachers claim they have no time to prepare multimedia slides and web pages, no time to learn PowerPoint. Students want multimedia; teachers don't prepare it. A digital divide separates a generation of computers from a generation of chalk.

How can we span this divide? These teachers brightened at the idea of a teaching assistant who would prepare these multimedia teaching materials, but only if the assistant could work from existing materials – typically typed course outlines – without supervision. They wanted a completely automatic teaching assistant.

So we created one.

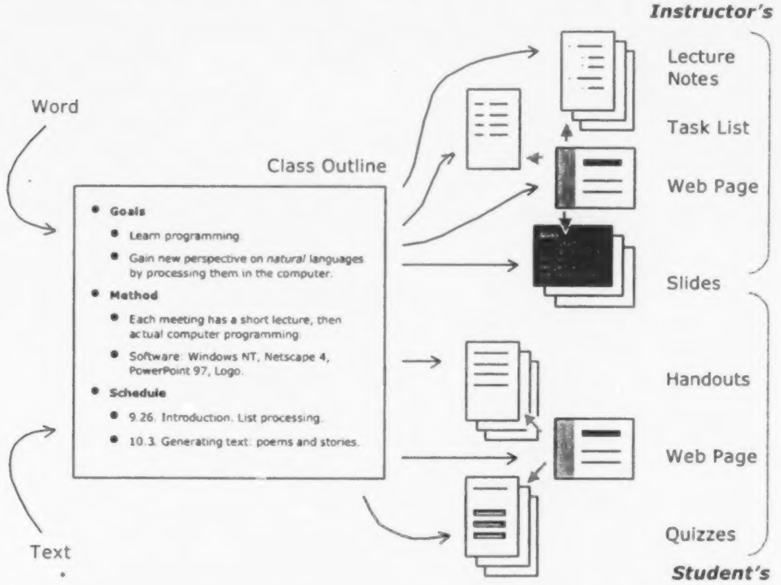
2 The Automated Teaching Assistant

The Automated Teaching Assistant (ATA) converts course outlines to multimedia teaching materials.

To use the ATA, instructors first prepare course outlines. They can use their favorite text editor or word processor, on any type of computer. Instructors then drag their saved file over the ATA program icon. The ATA reads the course outline and constructs a folder filled with teaching materials:

- syllabus (in the form of a class web page) [5]
- instructor' s notes (teaching plans)
- student' s handouts (outlines of each meeting)
- instructor' s task list (things to do, to prepare this class, sorted by date due)
- graphical slides visualizing each point in the outline [3, 4]
- web-based quizzes, tests, assignments, polls, class evaluations, and peer evaluations [1, 2]

The sketch below shows how a class outline is translated to teaching materials. Black arrows show the flow of information; **gray** arrows show hypertext links:



- **Goals**
  - Learn programming.
  - Gain new perspective on *natural* languages by processing them in the computer.
- **Method**
  - Each meeting has a short lecture, then actual computer programming.
  - Software: Windows NT, Netscape 4, PowerPoint 97, Logo.
- **Schedule**
  - 9.26. Introduction. List processing.
  - 10.3. Generating text: poems and stories.

All these materials are automatically uploaded to the class' s web server. Then students can view the materials from anywhere in the world. Instructors can travel to any classroom in the world with a working web browser, and give their lectures. There are no papers or floppy discs to carry, no worries about hardware and software compatibility, no need to install software, no fear that needed software will be missing.

The ATA is completely automatic: it has no commands or options. Teachers submit their outlines; seconds later the materials are all available on the internet. This automation contrasts sharply with the common manual production of multimedia materials.

If instructors were to create these teaching materials without the ATA, they would need to purchase and then study expensive and complex multimedia software, such as PowerPoint and DreamWeaver. In addition, they would have to learn at least some design theory, for they would need to learn how to make attractive and comprehensible slides, handouts, and web pages. (Although some instructors might find this an interesting diversion, others may resent it as time stolen from their research and content preparation.) Then, before every class, instructors must manually convert their lecture plans into slides and web pages. In our experience, this takes an average of four tedious hours to prepare each meeting. Most instructors, in fact, are unwilling or unable to spend this much time preparing lectures. So students are disappointed.

But if these instructors would use the ATA, it will cost them only seconds, but will greatly increase their student's satisfaction. The ATA is more efficient because it factors the *style* (layout and design) out of the *substance* (logic and content) of teaching materials. Instructors need concern themselves only with the creation of the abstract content of their classes; They can delegate the tedious physical layout and distribution to the ATA.

Using the ATA, we prepare lectures in an average of 40 minutes. The ATA allows us to prepare in only 17% of the time – it speeds preparation five times.

References

[1] Houser, C., "Introduction to Interactive web pages Using CGI", *Proceedings of C@LLing Asia 99* (1999).

[2] Martin, L., *WWWAssign*, http://www.northpark.edu/~martin/WWWAssign (1999).

[3] Meyer, E., *Designing Infographics*, Hayden, Indianapolis, Indiana (1997).

[4] Rabb, M., *The presentation design book*, Ventana Press, Chapel Hill, NC (1993).

[5] Weinman, L., *Designing web graphics*, New Riders, Indianapolis, Indiana (1996).

The Development and Evaluation of a Learning Support System for Converting Web Pages

KOYAMA Yasuhiro*, YOKOYAMA Setsuo, AKAHORI Kanji***
**Graduate School of Decision Sciences and Technology, Dept. of Human System Science,*
Tokyo Institute of Technology, 2-12-1 O-okayama, Meguro-ku, Tokyo,
TEL: 03-5734-3233
yasu@ak.cradle.titech.ac.jp akahori@ak.cradle.titech.ac.jp
***Information Processing Center,Tokyo Gakugei University*
yokoyama@cs.u-gakugei.ac.jp

In recent years, the use of the Internet for school projects has become popular, even in the primary level. One of the difficulties in the use of the Internet is the arrangement and integration of Web materials to meet the learner's goals. This paper presents a tool that will help meet this challenge. It will also describe how the tool was developed and what are the results of its evaluation. The features of this tool are the following: 1) learner can easily gather Web pages as thumbnail of a screen image; 2) learner can make a list of thumbnails; 3) thumbnails can be sorted, with comments added; 4) arranged thumbnails can be displayed by HTML. Further, the learner can make a presentation using thumbnails. The authors later conducted an experiment to verify the effectiveness of this tool in arranging Web pages. The developed thumbnail tool and the browser's bookmark tool were compared. The results showed that our developed tool was more effective than the bookmark tool, especially in following areas: (1) more recognizable contents of Web pages (2) easier operation, and (3) more user-friendly for students.

Keyword: **WWW, Exploring Projects, Screen Image, Thumbnail, Bookmark**

1 Introduction

In recent years, the use of the Internet for school projects has become popular, even in the primary level. In Japan, Ministry of Education will implement the integration of technology in K-12 starting 2002. Thus, students will need to have the skills needed when using the Internet for various school subjects. For project-based learning using the Internet, the popular tool for surfing and gathering online data will be the search engine. It enables the easy gathering of various online data. But not all online data is reliable and accurate. Also, if not updated, the data or information in web pages can become obsolete. So learners need a tool that will help collect, select, organize and integrate the web pages that meet their learning needs.

Currently, the tool that is available to learners is the bookmark tool. It enables users to save Web pages with its title. It also makes it easy to access the web site's URL. But the bookmark tool leaves much to be desired in terms of the organization and integration of online data. Because data gathering using search engines is a vast task, there is an immediate need for easy browsing. The bookmark tool is a tree-structured file system, which is not quite adequate for quick and easy browsing. Moreover, it is hard for learners to appreciate the significance of Web pages when they appear only as text names when bookmarked.

In addition to doing research projects, learners also engage in making presentations of their projects using the Internet. To help learners in this activity, the authors proposed a tool that will provide students an easy way of making a file for their presentation. So, the authors developed and evaluated a learning support system which will enable learners to arrange and integrate Web more effectively.

2 Conceptual Framework for Tool Development

To reduce the load on making our choice information , the following 2-part approach was taken:

(1) The centralized of system approach. In advance searches, the tool will automatically narrow down the search to the closest level possible (filtering approach). This means the goal is an intelligent tool that can select information and improve the precision of narrowing down the search.

(2) The centralized of human approach. By adding available information as hint, in order to reduce extraneous information. This a support to the select available information.

The overall goal of this 2-part approach is to enable an easy narrowing down of a search.

When gathering web pages for a school project using the Internet, the tool that was developed by the authors enables the capturing of web pages and viewing them as thumbnail images. The authors believe that thumbnail images are more effective in providing visual cues of the content of Web pages. And, by displaying thumbnails, learners can arrange Web pages holistically, that is, they can visualize the whole composition. The authors made the hypothesis that more visual information as that provided by thumbnail images will be more effective when arranging Web pages for a project or presentation.

For presentations, the popular tool is Microsoft PowerPoint. Compared to OHP presentations, the use of motion pictures and animation makes a presentation more dynamic. But for children who are beginning computer users, the use of such tools may not be easy or may require more technology resources than what is available. But, by converting web pages directly to a HTML coding for presentation, the learning curve will be lower. So the authors proposed to add the function of being able to integrate selected web pages into a HTML coding for presentation in the development of their new tool.

3 The development of the new tool

3.1 Overview of the new tool

The developed new tool enables users to arrange Web pages using thumbnail images (**Figure1**). The functions of the developed tool are: listing thumbnails, sorting, and scrolling. The added function of a memo or comment line is to enable the users to add new information or data. The developed tool will then automatically generate the HTML coding for presentations. Through the use of HTML, learner can easily make a presentation (**Figure4**). **Figure2** shows the system configuration. The procedure for the use of the developed tool is as follows:

1) Learner displays Web pages or self-produced HTML pages using Web browser.
2) Screen image of Web pages and page title are saved to a database.
3) Lists of thumbnail from the database are displayed. Learner arranges web pages on the display, and add own comments to thumbnail.
4) Finally, using the arranged materials, learner makes a simple presentation

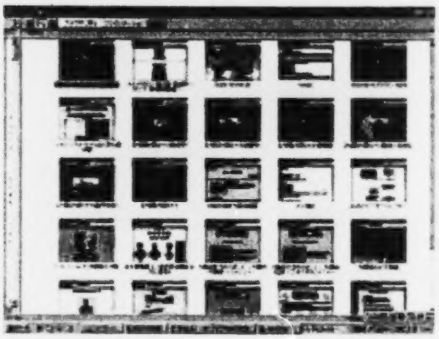


Figure1: A page showing the list of thumbnail images 1

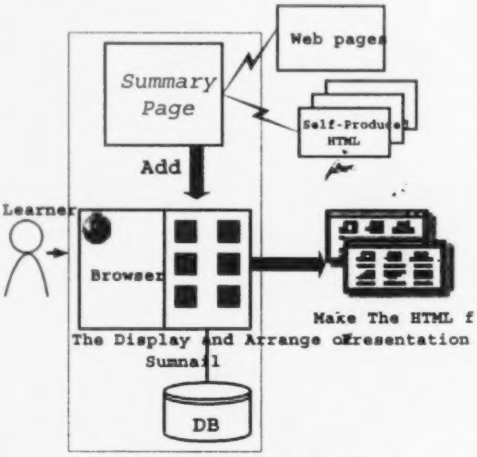

Web pages
Summary Page
Self-Produced HTML
Add
Learner
Browser
Make The HTML for Presentation
The Display and Arrange of Sumnail
DB

Figure2: System Configuration

3.2 The type of display Web page

In displaying the collected Web pages, the following 3 modes were used,
[1] Converting to thumbnail screen images
[2] Manipulating the original Web pages
[3] Making a presentation
The following sections explain further these 3 types.

3.2.1 Converting to thumbnail screen images

When selecting Web pages to put together, the user clicks a button to add a Web page. The web page is then converted to a thumbnail screen image (**Figure3**). Thumbnail screen images are Bitmap file made of large volume of data, so this Bitmap file is converted to a JPEG file. After that, the thumbnail is saved to the database.

3.2.2 Manipulating the original Web pages

By double clicking the thumbnail screen image, the learner can access the original Web page. It is just conceivable that learner will want to arrange the thumbnail web pages, and at the same time, have access to the original web pages. **Figure3** shows how the original web page and the lists of thumbnails are displayed at once. To change the display size, the learners can move from side to side, the display size control button located at the center of the display.

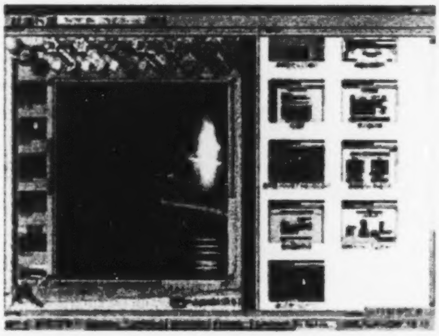


Figure3: A page showing the list of thumbnail 2

3.2.3 Making a presentation

Figure4 is the display of HTML for presentation. Arranged thumbnails are displayed in a sorted order. Learners can make a presentation using the display. Each Web page is composed of a link to the thumbnail, a link to the URL, and an area for comments or memo. The purpose here is to provide a function that will enable the easy arranging and integrating of Web pages for a presentation.

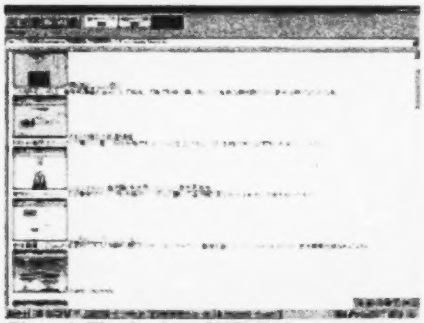


Figure4: The display of HTML for presentation

4 Evaluation of the tool

4.1 Purpose

The object of this evaluation is to verify the usability of the tool developed by the authors. Particularly, it will study the thumbnail screen images' usability for arranging Web pages. The subjects are the tool group using the developed tool and the bookmark group using only the regular bookmark tool. The groups were given the task to arrange Web pages about a specific theme. To collect data, the following were done:

(1) conduct a questionnaire survey. Subjects evaluated the operationality of the tool and were asked to give written comments of their experience of using the tool.

(2) In terms of arranging web pages, users compared the tool with the bookmark tool, and the analyses of the following data items were done.

1. work time
2. total number of times a URL is accessed
3. number of times a URL is re-accessed (the same Web page is accessed more than 2 times)
4. number of times thumbnails are sorted
5. number of times thumbnails are deleted

4.2 Method

The subjects arranged Web pages based on a theme using the developed tool and the bookmark tool. Thirty (30) Web pages were prepared in advance by the experimenter. To get a history of how they operated the tools (history of operation), a video record of how the subjects used the tool was made from a TV converter to a VHS video tape. After the experiment, the subjects answered the questionnaire. The experiment had the following stages

1. The use of the developed tool and the bookmark tool was explained to the subjects;.
2. The content of the task (theme of project) was explained to the subjects
   Theme A: the sights of Tokyo that you want to introduce to friends
   Theme B: the sights of Osaka that you want to introduce to friends
3. To eliminate order of effect, the subjects were divided into 4 groups (Table 1).

	Former	**Latter**
1Group	Theme A Using the tool	Theme B Using bookmark
2Group	Theme B Using the tool	Theme A Using bookmark
3Group	Theme A Using bookmark	Theme B Using the tool
4Group	Theme B Using bookmark	Theme A Using the tool

Table1: Subject groupings in the experiment

4.3 Results

To compared the developed tool and bookmark tool, questionnaire data was analyzed for significance using the t-test The results are given in **Table2**.

	Average	**Result**
Thumbnail screen image is more recognizable	**4.17**	**
The lists of thumbnail are more recognizable	**4.17**	**
Useful for arranging web pages	**3.83**	*
Recognizes the contents of a web page	**4.33**	**
Useful for school projects that use the Internet	**4.67**	**
Useful for making a presentation	**4.42**	**

*p<.05,**p<.01 t-test (two-tail test) the average(max5)

Table2: The results of the questionnaire

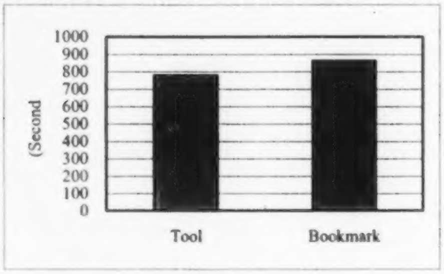
T-test results show that web page titles with thumbnails are more recognizable than text-only web page title. And as to browsability, the lists of thumbnail are more recognizable than the tree structure of the bookmark tool. Inquiry as to "useful for arrangement" was significant at the 0.05 level. But as to the ability of operation in the questionnaire, couldn't get level of significance. Because the interface of sorting the thumbnails will not be enough to good for learner.

In the analyses of the history of operation (reference 4.4 (2)), the record shows that the thumbnail screen image is useful to learner when arranging web pages. The results are indicated in **Figure6-10**. From the results, the following items were verified:

* For shorter work time, the developed tool is comparatively more efficient than the bookmark tool (Figure6).

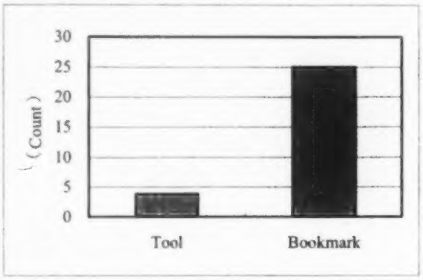
* By using the thumbnail screen image, the learner is able to better recognize the contents of the web page (Figure7,8).

* Lerner is comparatively able to estimate whether to use web pages or not (Figure10).



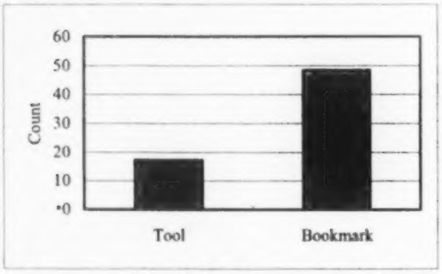
+$p<.1$

Figure6: Comparing the average of work time



**$p<.01$

Figure8: Comparing the average number of times of re-accessing URL



**$p<.01$

Figure7: Comparing the average of the total number of times of accessing URL

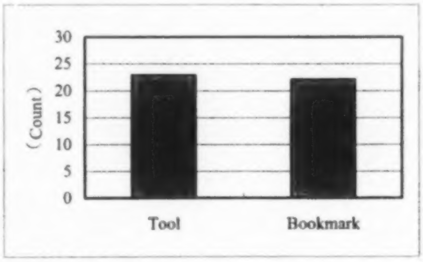
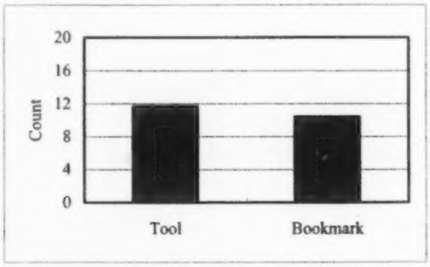


Figure9: Comparing the average number of times of sorting thumbnails



+$p<.1$

Figure10: Comparing the average number of times of deleting thumbnails

4.4 Analysis

The results of the evaluation procedures show that
1) based on the questionnaire, there were good results as to the functionality of the thumbnail screen images. And from the subjects' comments, "the lists of thumbnail is useful", "helps better recognize contents of the web page", and "the arrangement of web pages using the tool is convenient and useful".
2) based on the results of history of operation, work time, in terms of the number of times of accessing and re-accessing the URL and the number of times of deleting thumbnails, got good results in the given level of significance.

In terms of browsability, providing the user with a list of thumbnail is more useful than the bookmark tool. Accordingly, for arranging web pages, the list of thumbnail was better for integrating the collected data and for reviewing them. For arranging web pages, the results of the history of operation show that the developed tool is more useful than the bookmark tool.

5 Conclusions

In this research, a tool for learning to support the arrangement and integration of web pages was developed and evaluated. The results of the study can be summarized as follows:
1. Development of the learning supporting tool
   This research addressed the problem of selecting information for research projects using the Internet [1.Introduction], and examined how to resolve the problem by developing a tool that is both effective and user-friendly. The research also considered the interface of the tool and provided a conceptual framework [2.Conceptual Framework for Tool Development] in its development.
2. The evaluation of subjects about ease of operation and usefulness of the tool
   In the experiment phase of the paper [4.The evaluation of tool], a questionnaire was used to measure the as to ease of operation and usefulness of the tool., and got good results.
3. Verifying the efficiency of the tool for manipulating web pages
   When it comes to accessing and re-accessing URLs, the tool was more useful than the bookmark tool. For arranging web pages, the availability of a list of thumbnail images made it easier to integrate the selected web pages and to review them.

5.1 Future Studies

For future studies, the following are recommended:
1) Modification of the tool and adding more functions
2) A detailed analysis of the operation history

Acknowledgement

The authors would like to thank Dr. R. Santiago for her great assistance.

References

[1] Akai, K. and Koike, H., "Experimental Studies on Visual Search of Image Files" (in Japanese), The Special Interest Group Notes,: Information Processing Society of Japan(IPSJ), Vol.98, No.35, 98-H1-78, pp.27-32(1998)

[2] Ueda, M., "Development of a system of learning support using the Internet" (in Japanese),: Japan Society for Educational Technology(JET), 99-4, pp.9-16(1999)

[3] Mizobuchi, S. and Furuyama, T., "An analysis on the effect of indexes on WWW information selection process" (in Japanese), The Special Interest Group Notes of IPSJ,: Information Processing Society of Japan(IPSJ), Vol.98, No.116, 98-H1-81, pp.31-36(1998)

The Estimation of Music Genres Using Neural Network and Its Educational Use

Yasuhiko TSUJI*, Kanji AKAHORI*, Atsuhiro NISHIKATA*
Graduate School of Decision Science and Technology, Tokyo Institute of Technology
2-12-1, O-okayama, Meguro-ku, Tokyo, 152-8552, Japan
Tel: +81-3-5734-2991
Fax: +81-3-5734-2991
E-mail:tsuji@ns.cradle.titech.ac.jp,
akahori@cradle.titech.ac.jp,
nisikata@cradle.titech.ac.jp

To develop a learning support system of music genre, a neural-network-based system was developed that can estimate the genre of music from partial information of a standard MIDI file of music. Standard MIDI files of 120 music titles has been identified into 4 genres, Japanese Popular Ballad, Jazz, Hard Rock and Heavy Metal after the Neural network of the system had been trained. Comparison shows that, the system developed, has a higher judgment rate than that of subjects. Next, the weight of the links were examined by an expert, 5 of the nodes in the Hidden Layer could be extracted.

Keywords: **Music Education, Neural Network, Intellectual Learning Support, MIDI**

1 Backgrounds and Objectives

Recently, popular music, for example *Beatles* etc, is included in recent music textbooks of Elementary, Junior High and High Schools in Japan. So, it is thought that music education using popular music will increase more and more in course of time. When students learn popular music, music genre of the music is an important factor[1]. In order to learn the musical feature of each genre, it is thought to be very effective. Systematic genres studying of popular music, in which students seems to be interested, is thought to be a way of the students' music experience enrichment.

An "Automatic Composition MAGIC (Music system for Arrangement and Intelligent Composition) Considering Music Style" was developed [2] by Minamikata in 1989 is one of the researches in the research field that treats plural genres of popular music. This System supports composition and adaptation using heuristic rules divided by music taste of genre. It is said that rule-based system like this is effective when the system reproduces a already-known music taste or rule for the system, but there is an anxiety that generated music is conventional, and it is a problem for an unknown taste.

It can be said that the genre of popular music is the combination of different music. Now, many researches have done the grouping of music. Concerning Neural Network-based research, the research of Sakamoto (1999) grouped the music according to the sensibility information by using SD method [3]. If consider the flexibility and generality grouping by neural network differs from that of grouping by rules or multiple different analysis. So, it is said that moderate result can be expected for any unknown input by the process of grouping by Neural Network.

Based on the above research, we aim to develop the learning support system which can provide feedback on "Feature as the genre" of an unknown music with the Neural Network training of the music of various genres. Based on the above-mentioned background, we conducted this research in the following way. At the beginning, reserve experiment was done by an expert of popular music to confirm the factor for the estimation of the genre. Based on the obtained finding, we trained the Neural Network. Here the Neural Network was composed using the partial information as input signal and genre of the music as output or teaching signal. In order to use this system for education purpose in the future, the meaning interpretation for each factor of the Hidden Layer of the trained Neural Network was identified by an expert of popular music. Then, the genre estimation experiment was done using the subjects who seemed to have general experience of popular music. Lastly, the estimated average result of the subjects and the estimated result of this system was compared to show the effectiveness of this system.

2 Estimation of Music Genres by Expert

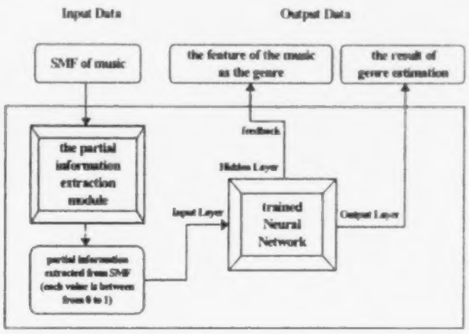
When music and genre are trained to the Neural Network, the problem is that we should take data to make an input signal from a long standard MIDI file. Therefore, we examine the mounting method of this system by knowing how the person judges the genre. For that, in the preliminary experiment we ask the expert about the factor of the genre estimation. The subjects had different musical instrument performance experience for ten years or more. The procedure was that they were made to listen ten in total of five genres. Also the factor to estimate the genre was interviewed. As a result, the following factors were found.

(1) The factor to estimate the genre is various according to the genre, and it's vague information.
(2) The factor to estimate the genre is local & partial information.

From (1), at first we got to the hypothesis that the estimation of music genre based on rules is very difficult and not proper. Under the above hypothesis, we propose to use Neural Network to deal with vague information in this research. As the input from (2), we judged that it was appropriate to extract partial information that seemed to be necessary for estimating the genre of music, and to assume it to be an input value of the Neural Network. The standard MIDI file (Hereafter, it is abbreviated as SMF) that is already a descriptive language was used as music.

3 Genre Estimation System

Figure.1 shows the composition of the genre estimation system. The flow of this system is as follows. When the user inputs SMF of music, the partial information extraction module extracts some partial information from the music. Then, it is put to the Input Layer of the Neural Network that has already been trained for music and the genre. The Neural Network feeds back the result of estimating the genre obtained from the Output Layer. Moreover, the feature of the music as the genre obtained from the Hidden Layer is planned to use as feedback in the future. If the module is developed, the user will be able to learn the genre.

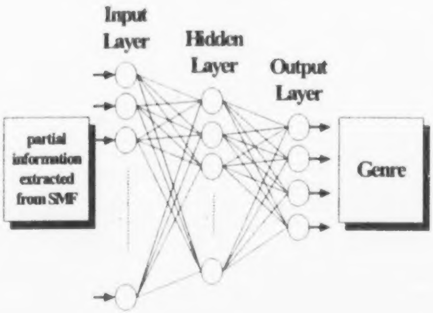

Input Data
Output Data
SMF of music
the feature of the music as the genre
the result of genre estimation
feedback
Hidden Layer
the partial information extraction module
Input Layer
trained Neural Network
Output Layer
partial information extracted from SMF (each value is between from 0 to 1)

3.1 Extraction of Partial Information from SMF

SMF of the General Midi correspondence was used in this research. SMF includes various musical information such as Note-On (time of starting to ring each music sheet), Note-Off (time of finishing to ring each music sheet), Velocity (the strength of each attack), Note Number (pitch), and Program Number (kinds of musical instruments and tones) etc. The following three information of these score information were decided to use in the partial information extraction module.

BEST COPY AVAILABLE

1. Kind and tone of musical instruments extracted from Program Number (henceforth, we call it "Musical Instruments and Tones", which is expressed by an array of 128 Boolean type variable. Each valuable shows whether musical instruments (tones) of Program Number 1-128, were used in that music or not.).



Distribution of Rhythm extracted from the statistics of position of Note-On per a bar (henceforth, we call it "Distribution of Rhythm", which is expressed by an array of 16 integer type variable. Each variable shows the frequency for which Note-On event is held at the rhythm in one bar in the SMF.).

Distribution of Pitch extracted from the note number (henceforth, we call it "Distribution of Pitch", which is expressed by an array of 12 integer type variable. Each variable shows the frequency for which each pitch of 12 music scales is used in the entire music of SMF.).

3.2 Composition of Neural Network

Figure 2 shows the composition of the Neural Network. We adopted the Back-Propagation algorithm as the learning algorithm of the Neural Network. For the input signal, we used a combination of the values.

4 Outline of Genre Estimation Experiment using this system

4.1 Method

By the above-mentioned methods, the genre estimation experiment by this system was performed. 120 music titles of SMF which are composed of 30 titles each in Japanese popular ballad, Jazz, Hard Rock, and Heavy Metal, tried to be learned by the Neural Network. In this research, the combination of the following partial information was learned as an input data.

Musical instrument and tone	128bit
Distribution of rhythm	16bit
Distribution of pitch	12bit
Musical instrument and tone, Distribution of rhythm (+)	128+16=144bit
Distribution of rhythm, Distribution of pitch (+)	16+12=28bit
Musical instrument and tone, Distribution of pitch (+)	128+12=140bit
Musical instrument and tone, Distribution of rhythm, Distribution of pitch (+ +)	128+16+12=156bit

The number of units of Hidden Layer in each Neural Network is assumed to 10-30. The number of units of Output Layer is as many as the number of genres that the Neural Network learns. In this case, it requires four units in Output Layer, because there are four genres.

4.2 Result

NN	Input Layer	Hidden Layer	Output Layer	Judgment Rate	Judgment Percentage	Learning Time
	128	20	4	119/120	99.2	X
	16	10	4	97/120	80.1	X
	12	10	4	103/120	85.8	X
	144	30	4	120/120	100	About 650
	28	30	4	111/120	92.5	X
	140	30	4	119/120	99.2	X
	156	30	4	120/120	100	About 1100

The result of training is shown in Table 1. In the Table 1, "NN" means Neural Network, and - in NN means the Neural Network whose input information is described above. The result of training, NN was converged about 650 learning times, and

BEST COPY AVAILABLE

NN was about 1100 learning times, but other NN were not converged within ten thousand learning times. So, the trained Neural Network was able to judge the genre of learned music at 100%.

From this, it is suggested that the Neural Network like - that has single partial information in Input Layer can't finish learning, But the combination of those partial information make it enable to learn. This result supports the findings of experts at the preliminary experiment in Chapter 2 whose also says that the factor to estimate the genre is various according to the genre.

4.3 An Analysis of Hidden Layer

The Hidden Layer in the Neural Network is analyzed here. There is a heuristic method that each cell's tendency in which it is likely to make active or inactive is found by an expert, and then the meaning of factor is obtained[4],[5]. We used that method here.We focused on the weight of the link between Hidden Layer and Output that is above 10. Each unit from No.1 to 5 are activated by following genres.

Unit No.1: Hard Rock
Unit No.2: Hard Rock, Jazz
Unit No.3: Hard Rock, Jazz, Japanese Popular Ballad
Unit No.4: Heavy Metal
Unit No.5: Japanese Popular Ballad

Table 2. Analysis of Each Factor in Hidden Layer

Unit	Name of Factor
1	**Hard Tone Factor**
2	**Synthesizer Tone Factor**
3	**Jazz-Acoustic Factor**
4	**Rhythm Tendency Factor**
5	**Combination Factor of Electric Instruments and Rhythm Tendency**

Finally, each unit was named by a music expert. The summarized result is shown in Table 2.

5 Experiment by Subject

To investigate at how much rate can the subjects, twenty-five female university students were asked to listen to eight music titles of 4 genres of SMF with MIDI sound randomly, and to judge the genre and the factor for each music.The judgment rate of all the subjects was 66.5%.

To compare the judgment of subjects with this system, Neural Network was trained with 119 titles, and was made to estimate the genre of subtracted one as unknown music.

As a result, both Neural Network and have a judgment rate of 100% for eight unknown music titles. From this, the judgment of this system is higher than that of subjects with general experience of popular music.

6 Summary of Results

In this research, development and evaluation of genre estimation system ware performed aiming for the development of learning support system of music genre. The results are summarized as follows:

(1) The preliminary experiment for experts with an experienced popular music was performed, and a result that says that the factor to estimate the genre tends to be local & partial information was obtained.

(2) From this finding, genre estimation system using Neural Network was developed.

(3) 120 music titles have been identified into 4 genres, Japanese Popular Ballad, Jazz, Hard Rock and Heavy Metal at the rate of 100% by training the Neural Network to identify these 4 genres.

(4) The judgment rate was 66.5% as the result of the estimation experiment for subjects with general experience of popular music.

(5) This system was made to estimate 8 music titles, as an unknown music, out of 120 which were used in the genre estimation experiment by subjects. As a result, the estimation rate of 100% which is higher than that of the subjects (66.5%) was obtained.

(6) Each unit of Hidden Layer in trained Neural Network was enable to be named, and the factors of each unit were able to be extracted by the expert of popular music.

From this finding of 6, providing feedback on the features of the music from Hidden Layer becomes possible by the way of observing the result of meaning explanation of Hidden Layer in which the Neural Network has the feature of the music as a genre, observing the state of fire, and observing the input units which have tendency to make active to the fired units in the Hidden Layer.

From the result described above, the possibility of the development of a learning support system using this system for music genre is shown. And, it was thought that the trained Neural Network of this system has the application possibility not only to the learning support system but also to the supporting composition and adaptation.

Acknowledgement

The authors would like to thank Mr. UZZAMAN MD. ANIS and Assistant. Taizan SUZUKI for their cooperation.

References

[1]Hiroshi UEDA, Kazumi AKAIWA etc.: Hajimeteno ongakutsuu:50 genres 500 CDs, Ongakunotomosha,1993

[2]Junichi MINAMITAKA, Mayumi INO, Kunio SATO, Shigenori MORIKAWA: Automatic Composition MAGIC Considering Music Style, Collected lecture papers of the 39-th National Convention of Information Processing Society of Japan, No.3, pp.2242-2243,1989

[3]Takashi SAKAMOTO, Yoshinobu KAJIKAWA, Yasuo NOMURA: Classification of Group According to Tune's Impression Using Nonlinear Distinction Analysis in the Music Sensibility Space, Transactions of Information Processing Society of Japan, Vol.40, No.4, 1999

[4]Hisayoshi ITO, Hiroki ITO, Kanji AKAHORI: An Analysis for the Neural Network Supporting Users' selection of Menu Items,IEICE technical report, (Educational Technology) ,Vol.96, No.500, pp.53-60, 1997

[5]Michio MATSUBARA: Evaluation of Global Processing of Students in Electric Circuit, Transactions of Society Japan Science Teaching, Vol.39, No.1, 1998

The Usability Aspects of an Universal Brokerage and Delivery System for the Pan-European Higher Education

Borka Jerman-Blažic
University of Ljubljana, Faculty of Economics and Jož ef Stefan Institute
Jamova 39, Ljubljana, Slovenia
Borka@e5.ijs.si

This paper gives short overview of a recently launched EU project for universal exchange of university course units for the higher education based on a brokerage and delivery system model. The basic components and characteristics of the approache are described. More attention is given to the approach that will be used for assesment of the innovation and for evaluation of the learning and educational achievements.

Keywords: **distance learning, brokerage and delivery system**

1 Introduction

The new technologies are transforming the ways business operate and the ways people work. They are also reshaping the expectations, needs, and opportuniies in education and learning. The customers of the education market are forcing the education to become demand-led, rather than production driven. The current technologies are providing basis for a new just in time, on demand approach the electronic educational products to be offered by virtual or classical universities through special platforms acting as intermediaries between the consumers and the suppliers of educational material. It is clear that the new tecnology alone will not make this new model of more efficient education to happen by itself. Rather, new innovative models of production, brokerage, delivery and presentation are needed that will put together the participants of the education process to collaborate globally and to use the advantages of the powerful technology.

The European project »UNIVERSAL-brokerage and delivery system for exchange of university course units for the Pan European Higher education« [1] is one of the attempts this model to start to work. The project is part of the Vth Framework program of User friendly information society [2] funded by the Commission of European union. The project started in March this year and will last 3 years. The conzortium of the project is large as it encompassess 17 partners from EU[3], among them twelve higher education instituitions from all over Europe, two research organisations, one SME[4] and two telecom service providers. The consortium has also partners from outside Europe as the model developed is expected to have global applications. These are: the Moscow research institut, HEC from Montreal and Nanyang Technological University from Singapore.

The model and the implementation in UNIVERSAL is based on an education brokerage and delivery system being developed to incorporate training by provision of valued-added services to both the customers of the system and the suppliers of the educational material. The major characteristic of the system is offer of different types of learning and delivery of the educational material and its distributive nature. There is no central site for delivery of educational material. Additional characteristic of the model is the system that will be developed for pan-european acreditation of the purchased and performed university course units by the participants belonging to institutions of the European higher education system. At this early stage of the project is not possible to predict all expected circumstances and consequences that such system may have in the furure development of the educationa nad training. Brokergae approach in provision of distance education must first address a number of technological and educational isues which are part of the UNIVERSAL work program. In depth discussion of the pedagogical paradigms that may take place as well as te number of technological issues of the work program are not possible to be discussed in this article because of space restriction. Here, we concentrate on the basic characteristic of the model; highlithing the implementation scenario and giving more information about the usability aspects of the technology that will be evaluated through monitoring of the educational.

---

[1] UNIVERSAL -IST-1999-11747
[2] URL:http://www.ispo.cec.int
[3] EU stands for European Union
[4] SME- Small or Medium Size Enterprise

2 The model and the architecture of the system

The model being developed is based on creation of an open, cross-border, educational market environment coupling brokerage and delivery of "live" and "packaged"[5] courses. The proposed framework accommodates and adds value to the various business models and course structures employed in European HEI[6] institutions. It will enable:

- a single faculty wishing to experiment with the simple import of external material to enrich a specific course,
- an existing alliance between institutions to make their exchange more efficient and to enrich it with types of course units not previously exchanged,
- an Open University to extend the range and depth of its courses.

The UNIVERSAL brokerage platform is an interactive hypermedia environment offered to the academics and administrators of European educational institutions to plan and select courses. It de-couple offers and course units provision on the supply side from enquiry, booking and delivery on the demand side. The most important element that enable this de-coupling is the catalogue of offered educational material and the supporting processes, that adequately describes all the properties, educational and technological of the course units. This approach is implemented as brokerage platform and a number of delivery platforms, see Fig.1.

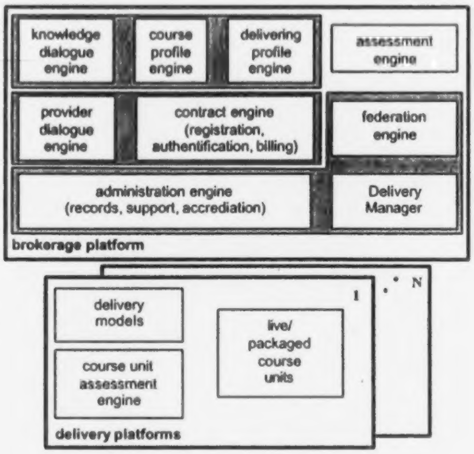

knowledge dialogue engine
course profile engine
delivering profile engine
assessment engine
provider dialogue engine
contract engine (registration, authentification, billing)
federation engine
administration engine (records, support, accrediation)
Delivery Manager
brokerage platform
1
N
delivery models
live/ packaged course units
course unit assessment engine
delivery platforms

Fig.1 The general architecture

The brokerage platform is further divided in a customer part, a provider part and an administration part. In the customer part, a knowledge dialogue engine is responsible for the dialogue to the demanding institution or to the enquiring student. When interacting with customers, the dialogue engine establishes their background knowledge and guides them in the selection of a course unit. Prospective customers are presented with choices according to a) pre-requisites and conditions attached to different types of courses (course profile engine), b) the suitability of different Institutions offering courses and c) the different delivery modes available for a particular course (delivery profile engine). Students will be able to make inquiries and express interest in course units that their home institutions have pre-selected and are willing to recognise and give credit. [7]In the provider part a provider dialogue engine is mainly used for feedback from the system back to the content provider. It is also used to handle the offer of content provision to the brokerage system to help academic and administrator users to plan new courses, submit course units, obtain customer records as well as learning progress profiles and assessment results. Demands for course units are sent to the administration engine, which looks for, offered course units fitting into the demand profile. The contract engine sets up the contract between offering and demanding side, thus it is dealing with registration, authentication and billing. In the administration part the system is driven by the administration engine, which is mainly a service database combined with a search engine. The tasks of observing and managing course scheduling and delivery issues, as well as timed interactions between the system and customers to enable the provision of joint courses are the responsibility of the delivery manager. The federation engine is dealing with distribution issues of the engineering implementation, like consistency, caching, forwarding of requests.

---

[5] Packaged courses are multi-media textbooks and WWW based courses, live courses are CWCS broadcasted courses

[6] HEI stands for Higher Education Institution

For each class of course unit there is an appropriate delivery platform. Variations in delivery platforms are due to the nature of the interactions and differences in media content and formats. The project will implement a limited range of delivery platforms, linked to the brokerage platform, sufficient to show the potential of the model. Each delivery platform contains an inherent delivery model and specific elements dealing with the media content itself.

3 The technology applied

The technology used is ubiquitous, Internet based, offering common, portable solutions and large-scale, shared, cross-border systems. These systems include, but are not restricted to:
Brokerage system is WEB based with Internet links. It is a central point of information, control and administration and logically will be centralized which means that only certain parts will be distributed or redundant for reasons of performance and high availability. The brokerage system will essentially be an E-business system that uses technologies like XML, Java / RMI, CORBA / ODP traders or agent based systems. standard security technology and intrinsic service negotiation for content delivery. Advanced transaction and billing functionality based on AAA[8] security technology, implement sophisticated administration and monitoring interfaces to the delivery systems and the integration of assessment capabilities

Delivery system similarly to the brokerage platform consists of an existing product/technology plus some enhancements ("glue" or "shell" around it), that acts as mediator between the delivery system and the brokerage platform. Defining a unique architecture for this interfacing allows adding additional systems / products by just implementing the appropriate interfaces that plug into the general architecture. A delivery system within the general architecture (see Fig.1) consists of content source, network and content sink. The content source can be a standalone system or part of a broader platform. The same applies to the content sink. The following delivery systems previously tested and used will cover the required functionality: Non-real-time delivery systems: retrieval of non-real-time mono-media and multimedia contents (e.g. browsing through text and hypertext information, download of content files). Real-time A/V systems: Real-time retrieval of multimedia contents (stored and live contents including A/V material like MPEG-2 streams) in synchronous and asynchronous manner (on-demand, live and scheduled broadcast scenarios) using IP technology (unicast, multicast), and broadband technology (ATM, ADSL,). CSCW systems: Videoconferencing and Computer supported co-operative work (CSCW) to enable interactive forms of tele-lectures combining parallel transmission of A/V streams and course material (e.g. slides) with the possibility to interact with the lecturer as well as with other parts of the audience. This family of delivery systems shall also support real-time experiments, simulations and case studies.

---

[7] Scheduling of actual course units remains the prerogative of the institutions and booking of places on courses is considered to be an internal function of the institutions

[8] AAA stands for Authentication, Authorization and Access Control

All delivery systems are inter-working with the brokerage platform and content provider systems to synchronise announcement and content delivery, to guarantee controlled user access, and to manage selection, compatibility and resource usage in delivery. All supported delivery modes will be available as profiles of the A/V delivery family. This means that each content provider can easily select the appropriate profile according to the nature of his contents and his network resources. It is important to note that used the A/V delivery technology is not based on the current average bandwidth and quality availability of the ubiquitous Internet. It will offer real high quality of A/V contents as broadband communication over the Internet (with technologies like ADSL) for a relevant number of users is available in Europe as well as the broadband services emerging in the convergence technology market (broadband over Satellite or cable-TV, interactive digital TV services) are expected to boost the widespread usage of broadband A/V information in the global IT environment. CSCW technology applied follows the principle of standards compliance and openness as for example the powerful CSCW tool ISABEL, developed in the RACE and ACTS projects ISABEL and NICE, then the standards-based (e.g. ITU H.323) COTS products (e.g. MS NetMeeting), MBONE multimedia conferencing tools (VIC, VAT, WB) developed within project MECCANO etc.

4 Content provision and description

The general architecture and model envisaged as a point of interaction of "sellers and buyers" on one hand, and of "place of commerce for actual content" requires an intelligent abstract description of the contents. Therefore, meta-data standards for multimedia contents and for educational environments are applied in the catalogue building of course units. The meta data system used is based on the specification of the IEEE LOM 3.8 meta-data scheme[9] with some extensions relevant to the platform developed as for example: attributes that specify the copyright and IPR[10] protection, attributes that describe information about digital signatures, watermarking, attributes that describe the network requirements for provision of appropriate quality of service, attributes relevant to live content and attributes defining the type of the course unit which may be: packaged, live, CSWC or mixed. The content provision and course description is based on a meta-dada system selected from available standard documents or previous projects results (1).

Several projects, that have investigated the management of information retrieval and the utilization of metadata for education and training have already proposed sets of meta-data requirements, like the Instructional Management System (IMS) project in the US (2) or the GESTALT (3) project in Europe. Some of the proposed sets have been evaluated and selected for the use in the UNIVERSAL project.

They are:

- **Learning resource content meta-data:**

Learning resource content meta data that enables cataloguing of contents of arbitrary aggregation level. UNIVERSAL supports the following granularity levels: *Fragment (Course Unit)*, *Lesson*, *Module* and *Course*. Each learning resource submitted by the "seller" is classified according to the aggregation levels and is added to the UNIVERSAL catalogue. UNIVERSAL supports "packaged" learning content, which is asynchronous in nature and synchronous learning content. Synchronous learning content is delivered as live transmissions of lectures, optionally supplemented by synchronous group ware communication technology. The special or unique features of the live content is described by specially developed attributes that are not part of the current existing meta-data standards.

- **Course structure meta-data:**

The UNIVERSAL brokerage platform not enables "buyers" to locate, use and re-use single course units. A functionality of the brokerage platform enables combination of single course units into higher levels of aggregation e.g. for full subject. This allows production of "custom" tailored complete courses. This is reflected in the course structure meta enabled by Course Structure Format (CSF) defined by the AICC and the ADL (1).

---

[9] IEEE Learning Objects Metadata document by WG 12 of IEEE Learning Technology Standardization Committee, URL:http://ltsc.ieee.org/wg12/scheme.html

[10] IPR stands for Intellectual Property Rights

- **Contents packaging meta-data:**

Like many other LMS, the UNIVERSAL platform offers the possibility to access packaged course units, which are mainly pre-recorded. Packaged courses enable an interaction with the contents itself, the interaction with the lecturer is usually not offered. UNIVERSAL packaged courses are described according to well-defined rules, specified e.g. in available standards like the IMS CPS (Content Packaging Specification).

- **Descriptive information about assessment procedures:**

Assessment is an important concept in on-line education and learning. Although the UNIVERSAL platform is not designed to assess the student' s advancement and the learning achievement, it will enable consumers or "buyers" to assess the functionality and usability of the platform and the delivery process. The assessment is performed with on line involvement of the students. Student' s responses will be captured using some form of structured mechanism with designed template(s) for such purpose. This is especially required in the cases when questions in the templates involve multiple choice, matched items, text selection, etc. Several structures exits for this purpose already: QML, SATML, and the IMS "Question and Test Interoperability Information Model" (QTI), UNIVERSAL team will decide which of the proposed template will be used for particular part of the platform and the delivery system used.

- **Meta-data for synchronisation:**

The asset management is important if the system is built up from re-usable learning resources e.g. units of lower aggregation level as is the case with UNIVERSAL catalogue. . In such cases the lifecycle management of the unit must be supported e.g. a component is not deleted from the asset management system in cases when the course unit is added to a module or larger course. This property of the system is described within the meta-data for synchronisation.

The UNIVERSAL consortium brings together a selected group of engineering and business schools across Europe who are interested in pooling their academic resources for the purpose of broadening the choice of courses and pedagogical materials offered to students and teachers and to provide opportunities for international collaborative learning. Although a few of its members have already established bi-lateral academic exchange programmes, in the initial phase of the project, partners are collecting course units developed within particular partner or outside in order to build a catalogue for the greatest market potential for the brokerage platform. The catalogue contains in addition to the meta-data information also: brief description of the educational objectives of the academic content for the students to whom courses are offered. In addition to that, brief description of teaching methodology used at each institution, description of the academic calendar for each institution, and description of the academic accreditation process for each institution are also information provided in the catalogue. Currently course units are collected from the following fields: Introduction to Information Systems, Regional Economic Development and Telecommunication, Global Marketing Management, Business Case studies, Metallurgy simulations and experiments, Foreign Languages and Cross-cultural Behaviour, Statistics for Economist, Tele-management.

The UNIVERSAL brokerage platform enables to locate, use and re-use single course units and combine single course units into higher levels of aggregation. This approach makes possible a definition of custom tailored structure for a complete course, which will lead, to a definition of a standard system of granularity for learning resources with the other projects from the IST cluster "Flexible University".

Several institutions already made suggestions for a hierarchy of aggregation levels. The IEEE LTSC and the IMS so far define four levels of granularity: *Fragment*, *Lesson*, *Module* and *Course*. Some other US based systems of aggregation can be found in the bibliography enclosed (8).

A course structure representation defines all of the course elements, the course structure, and all external references necessary to represent a course and its intended behaviour. The ADL together with the AICC, IEEE and IMS have developed the so-called Course Structure Format (CSF) which was the recommended approach for the UNIVERSAL project. The CSF promotes reuse of entire courses and encourages the reuse of course components by exposing all the details of each course element. The CSF is intended to reduce or eliminate dependency of a course on a particular LMS implementation.

The CSF is also intended to represent a wide variety of course structures and content "aggregations".

Content structures can be represented by the CSF that range from very small "chunks" of content – as simple as a few lines of Hypertext Markup Language (HTML) or short media clip – to highly interactive learning content that is tracked by an LMS. The CSF is neutral about the complexity of content, the number of hierarchical levels of a particular course (i.e., "granularity"), and the instructional methodology employed to design a course.

The UNIVERSAL platform incorporates continuous assessment of content and the system itself based on the interactions between the customers using the delivery of course units and the system itself. This information will be used to improve the overall scheme and the content to teachers, administrators and other possible customers.

5 The usability and evaluation

"Evaluation is the activity that throughout the planning and delivery of innovative programs enables those involved to learn and make judgement about the outcomes of the innovation concerned "[11]. The UNIVERSAL project aims also to develop tools to monitor the innovation process of education and learning and to develop best practice guidance.

The assessment of the content and the overall system, components for functional assessment are incorporated in both the brokerage system and in each of the delivery systems. Results from previous projects including the deployment of trans-national multimedia learning schemes[12] have shown that it is vital that all participants involved in the creation of the exchange platform and its educational content have a mutual understanding of the platform' s operations, functions and of the components' interaction. To fulfill this goal it is necessary to give the users the tools enabling an easy the use of the exchange platform such as: an administrator guide, oriented towards the management of the platform, from a technical ("how to use") point of view as well as from a content ("what to do") point of view; a user guide, describing the day-to-day use of the system e.g. to a teacher wishing to use content available through the platform (how to access the catalogue, how to book a live course, etc). In addition, in the case of "live" delivery systems (on-line live courses with CSWC), experience have shown that it is necessary to organise "hands-on" training sessions to free the teacher from the fear of new tools and to strengthen their "moderation" skills when working with a geographically distributed class through a TV-like systems. Classes in the live courses will be mainly cross-border meaning that most participants will not be working in their mother tongue and there will be a mix of cultures present in any one of the classes. As part of the preparation for participation in the main trials a short "Language & Behaviour" courses that will (I) help participants with their colloquial English (since the majority of the courses will be held in English) and (II) help them be aware of differences in cultural behaviour, e.g. questioning style will be provided.

The student/teacher ratio varies greatly among the UNIVERSAL consortium partners. A set of software monitoring tools are being studied to be implemented into the brokerage platform to make it possible for students to continuously assess their personal progress and to choose the academic path best adapted to his or her acquired knowledge and skills; for course unit providers tools will be used to improve the effectiveness of programmes offered to learners and modify content accordingly; tools will be used also to analyze the way learners use the courseware provided; to enable a global assessment of the usability of the platform etc. The monitoring tools as well as the usability evaluation techniques used for assessment of the innovation technology approach are being developed/selected in accordance with the ACTS Usability Evaluation Guidelines[13]. These guidelines define the testing and evaluation methods, experiments design, definitions of interviews, observations, heuristic evaluation and surveys.

The evaluation instruments for courseware evaluation and corresponding measures will include:

- Pre-task/post-task questionnaires
- Task experience questionnaires
- Computer experience questionnaires
- Exams or assessment of performance
- Post course questionnaires

---

[11] Stern(1988) in Evaluation of Learning Technology Implementation, by Barry Jackson, URL:http://ltdi.icbl.hw.ac.uk

[12] Project LEVERAGE AC 109 from the IV Framework Program-ACTS

[13] USINACTS – AC224, IV Framework Program -ACTS

- Knowledge quizzes
- Logs - logging times

The approach for educational assessment will follow the practical guidelines developed within the LTDI technology developed within the Learning Technology Dissemination Initiative funded by the Scottish Higher Education Funding Council[14]. As a result of this a set of guidelines identifying best practices for future users of an academic brokerage platform will be produced. Academic partners will have the possibility to review the existing experiences and pool their lessons learned from prior involvement in flexible, distance, and collaborative teaching and learning programmes. This information cover issues such as the choice and format for resources provided to students and teachers, access to tutors or teachers, methodology, independent study and collaborate work, learner motivation, learner monitoring, course accreditation etc.

6 Conclusion

The UNIVERSAL is a project that implements the EU policies regarding the development of the European higher education and the user-friendly information society in particular:

- By improving the quality and diversity of the pan-European HE system
- By promoting the globalisation of the exchange of HE course units
- By enabling partners from economically disadvantaged regions, particularly in Central and Eastern Europe to participate in these developments and helping them to strengthen and enrich their course offerings and the foster the education in general.

Most of the activity within the project will be tightly connected with the usability aspects of the applied methods and technology. Usability evaluation and proposed improvement will be based on the past experiences, guidelines and standards developed within projects that have addressed this issue of modern technology in depth. The consortium expects wide acceptance among the higher-level education institutions in Europe.

References

[1] Advanced Distributed Learning Network (ADLNet), *http://www.adlnet.org*

[2] Instructional Management Systems project, *http://www.imsproject.org*

[3] GESTALT:Getting Educational Systems Talking Across Leading-Edge Technologies, *http:www.fdgroup.co.uk/gestalt/*

[3] SCORM, Sharable Courseware Object Reference Model: *http://www.adlnet.org/Scorm/docs/scorm_2.pdf*

[4] IMS Question & Test Information Model, *http://www.imsproject.org/question/index.html*

[5] SATML Standardised Assessment & Testing Markup Language *http://www.engr.ucf.edu/aim/research/1998/tess/satml/default.htm*

[6] QML Question Markup Language, *http://www.questionmark.com/qml/index.htm*

[7] IEEE LTSC WG17: Content Packaging Working Group, *http://ltsc.ieee.org/wg17/index.html*

[8] IMS Content Packaging Specification, *http://www.imsproject.org/content/index.html*

---

[14] LTDI, Philip Crompton: Evaluation: A practical guide to methods and Stephen Draper: Observing, measuring or evaluating courseware URL:http://www.icbl.hw.ac.uk/ltdi

Use of abstraction levels in the design of intelligent tutoring systems

Ruddy Lelouche
Département d'informatique
UNIVERSITÉ LAVAL Québec G1K 7P4 CANADA
Tel.: (+1-418) 656 2131, ext. 2597 Fax: (+1-418) 656 2324
e-mail: `lelouche@ift.ulaval.ca`

In problem-solving domains (mathematics, physics, engineering, and most "exact science" disciplines), the knowledge to be acquired by the student is twofold: the knowledge describing the domain itself, but also and mainly the knowledge necessary to solve problems in that domain. As a result, an educational system in such a domain encompasses three knowledge types: the domain knowledge and the problem-solving knowledge, i.e. the knowledge to be acquired and mastered by the student, and the tutoring knowledge, used by the system to facilitate the student's learning process. In this paper, we show how these three knowledge types can be modelled, how they should interact with one another in order to fulfil the system educational purpose, and above all how abstraction levels can shed a uniformizing light on the system operation and make it more user-friendly. We thus hope to bring some contribution to the general and important problem of finding a generic architecture to intelligent educational systems.

Keywords: **Intelligent tutoring systems, Abstraction, Complexity, System Design**.

1 Introduction

Teaching is a very complex process in itself. Teaching strategies and activities vary considerably: by the role and autonomy they give to the learner, by the type of interactions they trigger with him/her, by the evaluations they enable, by the relationships they make between theory and practice, etc. From that last perspective, teachable domains can be classified according to the type of knowledge to be acquired by the student: "know", "know-how", and "know-how-to-be". Examples of such domain types are respectively: anatomy or a language grammar, the skill to solve a mathematical or medical problem, and the capability to adapt to one's environment or to deal with personal relationships. We are more particularly interested in the second type.

Moreover, almost all teachable domains vary in complexity, from simple basics to relatively complex problems to solve or situations to deal with. Thus, a student should learn and master the basics of such a domain before being taught wider notions. And when a human tutor detects errors or misunderstandings, s/he usually draws the student's attention on a small subset of the involved knowledge, so that s/he may correct his/her errors and/or misunderstandings, focusing either on a given set of the domain knowledge or on the scope of knowledge involved by a given problem.

Problem-solving (PS) domains are the ones in which we are interested here. In such a domain, the knowledge to be acquired by the student is twofold: the domain knowledge itself, but also and mainly the knowledge necessary to solve problems in that domain. As a result, an education-oriented system in such a domain, which we here call a PS-ITS, must encompass three knowledge types: the domain knowledge and the problem-solving knowledge, constituting the knowledge to be acquired and mastered by the student, and the tutoring knowledge, used by the system to facilitate the student's learning process.

This paper has two goals: to present each of the three types of knowledge involved in a PS-ITS, and for each type of knowledge, to show how abstraction and complexity levels appear and how we think it is possible to deal with them.

To do so, we present in section 2 our domain knowledge modelling and how we exemplify it in a few PS domains . Next, in section 3, we focus on the advantage of separating the problem-solving knowledge from the domain knowledge in a PS-ITS, and we present some problem-solving activities in various domains. In section 4, we briefly describe some principles of tutoring knowledge modelling in a PS-ITS. In each of these three sections, we show how to use abstraction and complexity levels, exemplifying them in a few typical domains.

Finally, section 5 presents the educational interests of using abstraction and complexity levels when modelling the three types of knowledge involved in a PS-ITS.

2 Domain knowledge

In order to describe the domain knowledge, we first present its characteristics in a general PS-ITS (section 2.1). We then show how we model it in a few PS domains (section 2.2), and how such an approach lets us introduce the notions of abstraction and complexity levels (section 2.3).

2.1 General

The first type of knowledge involved in every ITS, the *domain knowledge* (DK), contains all theoretical and factual aspects of the knowledge to be taught to the student. Although its specific structure can be varied, it typically may include concepts, entities, and relations about the domain [Brodie & al., 1984], object classes and instances [Kim & Lochovsky, 1989], possible use restrictions, facts, rules, [Kowalski, 1979; Clocksin & Mellish, 1981], semantic or associative networks [Findler, 1979; Sowa, 1984], etc.

The main system activities centred on this knowledge type are:

- providing the student with theoretical presentations and explanations about the various knowledge elements and their relationships in the teaching domain;
- providing the other modules of the ITS, i.e. problem-solving and tutoring, with the necessary background of domain knowledge that they need.

2.2 Application to a few domains

In the particular domain of **cost engineering**, Lelouche and Morin [1997; Morin, 1998] represent this type of knowledge with concepts, relations, and a special case of relations modelled as concepts, the factors.

Concepts can be basic entities like investment, interest, investment duration, present and future values, compounding, compounding period, interest rate, annuity, etc.

Concepts are linked to one another by various types of *relations*: either usual knowledge-representation relations, like *subclass of*, *element of*, *sort of*, etc., or numerical relations represented by formulæ. Such a formula is:

$$F = P \times (1 + i)^n \tag{1}$$

which, given the present value P of an investment over n periods at interest rate i, computes the corresponding future value F of that investment.

A formula such as (1) can be rewritten as:

$$F = P \times \Phi_{PF,i,n} \text{ where } \Phi_{PF,i,n} = (1 + i)^n \tag{2}$$

$$P = F \times \Phi_{FP,i,n} \text{ where } \Phi_{FP,i,n} = (1 + i)^{-n} \tag{3}$$

thus introducing the *factors* $\Phi_{PF,i,n}$ and $\Phi_{FP,i,n}$. Factors allow us to separate their definition (rightmost equalities above, a quantitative aspect) from their possible uses in the application domain (leftmost equalities, a qualitative aspect).

Similarly, the factor $\Phi_{AP,i,n}$ converts a series of identical annual amounts A into a unique present value P:

$$P = A \times \Phi_{AP,i,n} \text{ where } \Phi_{AP,i,n} = \frac{(1+i)^n - 1}{i\,(1+i)^n} \tag{4}$$

Actually, $\Phi_{AP,i,n}$ is a sum of Φ_{FP} factors (see details below). The factor $\Phi_{PA,i,n}$ does the reverse process:

$$A = P \times \Phi_{PA,i,n} \text{ where } \Phi_{PA,i,n} = \frac{i\,(1+i)^n}{(1+i)^n - 1} \tag{5}$$

There exist other factors converting gradient and geometrical series of amounts into a present or future value; such factors are also computed as a sum of $\Phi_{FP,i,n}$ factors.

In **geometry**, *concepts* are basic elements like point, line, segment, and later more elaborate elements like angle, then square, rectangle, circle, ellipse, polygon, solid, polyhedra, etc. Examples of *relations* between concepts are adjacency (of segments or angles), parallelism (of lines or line segments), complementarity (of angles), etc. Upper-level, more abstract concepts are then defined using lower-level ones, as well as relations between these lower-level concepts (e.g. a triangle is a set of three segments adjacent pairwise).

In **mechanical physics**, we similarly introduce *concepts* like time, distance, velocity, acceleration, mass, force, and later angle, angular velocity, angular acceleration, moment of inertia, torque, etc. We also introduce *relations* like the one defining velocity as the variation in distance per unit of time, or the one stating that the acceleration a is proportional to the force F that is applied. Introducing a generalization from linear to rotational movement, another relation defines angular velocity as the angle variation per time unit, and another one states that the angular acceleration α of a solid body is proportional to the torque τ that is applied to it. More precisely, we have:

for a linear movement $F = M \times a$ where M = total mass of the body (6)

and for a rotational movement $\tau = I \times \alpha$ where $I = \sum (m \times r^2)$ (7)

Equation (6) expresses Newton's second law. In equation (7), I is the moment of inertia and is expressed in terms of the mass m of each of its particles and of its distance r to the rotation axis. Obviously M in equation (6) and I in equation (7) play the role of factors as in cost engineering.

Although formulæ like (2-7) related to factors essentially involve quantitative aspects, the similarities and differences between them, and the circumstances regulating the use of either one, are of a deeply qualitative ground. In **cost engineering**, if the *value* of factors is indeed calculated from two or three numerical parameters, the *context* in which they are defined depends on whether we have to timewise move a unique amount or a series of amounts, identical or not, or conversely to compute an equivalent annual amount, etc. In fact, this context corresponds to the type of conditions that govern the investment, or *investment conditions type*, without respect to the amounts and durations involved, and is thus essentially qualitative. Similarly, in **physics**, the proportionality between force and linear acceleration, or between torque and angular acceleration, expresses a qualitative relationship. Only if the need arises, the exact relationship can be expressed by using the actual mass M in formula (6) or the result of the computation of the moment of inertia I in formula (7), which in the general case involves a simple or double integral. Indeed, did not the use of qualitative reasoning originate with qualitative physics?

2.3 Towards the notions of abstraction and complexity levels

In most PS-domains, abstraction most obviously appears in the *definition of the domain concepts* themselves, like we showed in all three domains above.

If *factors* are used in the domain, it also appears that every factor introduces an additional intermediate abstraction level between the concepts implied in the equation defining it. For example, in the case of formula (1), or equivalently formulæ (2) and (3) in **cost engineering**, or in the case of formula (6) and (7) in **physics**, we have (see figure 1):

- at the bottom of the hierarchy, basic concepts "making technicalities explicit" if necessary: the interest rate and the number of periods in cost engineering, the distribution of mass within the body volume in physics ;
- above them, concepts more fundamentally related to the problem being solved, namely in cost engineering the present and future values of the investment, and in physics the force and acceleration, or the torque and angular acceleration;
- between these two levels, an intermediate level created by the introduction of the factor (Φ_{FP}, Φ_{PF}, M, or I).

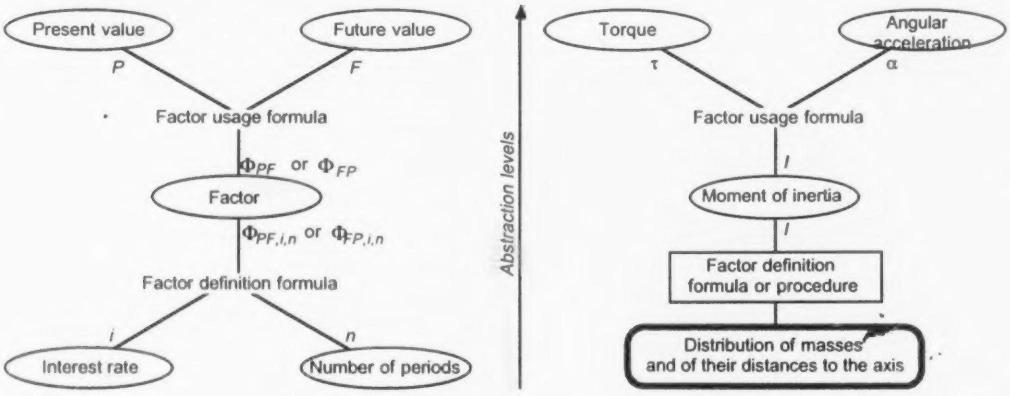

Present value
Future value
P
F
Factor usage formula
Φ_{PF} or Φ_{FP}
Factor
$\Phi_{PF,i,n}$ or $\Phi_{FP,i,n}$
Factor definition formula
i
n
Interest rate
Number of periods
Abstraction levels
Torque
Angular acceleration
τ
α
Factor usage formula
I
Moment of inertia
I
Factor definition formula or procedure
Distribution of masses and of their distances to the axis

Figure 1 — Representation of a factor as a concept.

That intermediate status of the factor, originally just an intermediate variable in computations [see formulæ (2) and (3) or (6) and (7)], makes it appear as a pedagogically oriented concept, which clearly separates

- the computational, quantitative aspect of the factor definition,
- from the practical, qualitative aspect of the factor usage in a domain problem.

This follows the theory [Lenat & al., 1979; Malec, 1989] according to which the use of multiple *abstraction levels* eases the modelling process and simplifies inferences which may be made on the domain concepts.

Most interestingly, our scaffolding approach can be made more general, at least in certain domains, where we may present and use higher-level factors built upon these first ones. Indeed, in cost engineering, "above" Φ_{FP} and

Φ_{PF}, the factors used to express the present and future values of a series of identical amounts (and vice versa) are a first way to generalize this concept hierarchy. For example, the Φ_{AP} factor is indeed a sum of Φ_{FP} factors:

$$\Phi_{AP,i,n} = \sum_{k=1}^{n}\Phi_{FP,i,k} = \sum_{k=1}^{n}(1+i)^{-k} = \frac{(1+i)^{n} - 1}{i\,(1+i)^{n}}$$

where the last expression results from computing the geometrical series shown. This example constitutes a proof of (4), but also and mainly shows that the Φ_{AP} factor is at a higher level than Φ_{FP}. Note that this refers to a *complexity level* rather than an abstraction level, since it is due to the way the Φ_{AP} factor is defined and computed. Similarly, the moment of inertia of a complex body can be (and often is) computed as the sum of elementary moments of inertia, and therefore is at a higher complexity level.

3 Problem-solving knowledge

In order to describe the problem-solving knowledge, we now present the general characteristics regarding problem-solving knowledge modelling in a PS-ITS (section 3.1). As we did in section 2, we then examplify our model in the cost-engineering and physics domains (section 3.2).

3.1 General

The second type of knowledge is specific to PS-domains [Ganeshan & al., 2000; Gertner & VanLehn, 2000], henceforth to PS-ITSs. We call it *problem-solving knowledge* (PSK). It contains all inferential processes used to solve a problem resulting from the instantiation of a practical situation based on the domain knowledge [Kowalski, 1979; Patel & Kinshuk, 1997]. In other words, in order to be able to solve a problem, the problem-solving knowledge needs a theoretical background, which is found in the domain knowledge. The processes stored in PSK may be represented in various ways, using any or all of: logic [Kowalski, 1979], procedural networks [Brown & Burton, 1978], semantic networks with procedural attachments, (augmented) transition networks, production rules [Goldstein, 1979; Anderson & Reiser, 1985], etc.

The main system activities centred on this knowledge type are:

- providing the inferential tools for problem solving, by the system or by a student;
- providing the inferential tools for coaching a student in a problem-solving session.

The main advantage of separating the problem-solving knowledge from the domain knowledge is that it emphasizes the distinction between the domain itself and the skills used to solve a practical problem in that domain, thus simplifying the learning process. That knowledge separation into DK and PSK is common to all PS-domains; this is why we believe that PS-ITSs, which are aimed at helping the student to learn how to solve problems, should display the same knowledge separation.

Besides, following [Lelouche & Morin, 1997], we can use — we believe in a novel way — that separation between DK and PSK to define four *generic operating modes* in a PS-ITS, based on the type of knowledge involved (DK or PSK), and on who "generates" that knowledge (the system or the student).

- In *domain-presentation mode*, the student asks the system some information about a domain theoretical element, and the system reacts by transferring to the student the required information or knowledge. The knowledge involved in this category is always DK, system-generated.
- In *demonstration mode*, the student asks the system to solve a practical problem or to coach him/her while s/he solves a problem. In the first case, the problem typically comes from the student him/herself, whereas in the latter one the problem typically comes from the tutoring system. In either case, the main level of knowledge involved is PSK, system-generated.
- In *domain-assessment mode*, the system prompts the student to develop a domain element, and the student thus expresses his/her understanding of that element. If judged necessary, the system may then intervene to correct that understanding. The knowledge involved in this mode is essentially DK, student-generated.
- Finally, in *exercising mode*, the system prompts the student to solve a practical problem. The student then solves it step by step, showing what s/he understands of the involved problem-solving knowledge and of the associated domain knowledge. If necessary, the system may decide to intervene in order to help him/her reach his/her goal or to correct it. The knowledge involved in this mode is naturally PSK, student-generated.

3.2 Application to a few domains

Several problem-solving activities are domain-independent, like:

1. identify and instantiate the given problem data;
2. identify and instantiate the expected result(s);
3. apply a formula;
4. apply a theorem.

Every PS-domain also has its own domain-dependent activities. For example, in **cost engineering**, we have:

5. draw a temporal diagram to represent the relevant events;
6. compare amounts located at the same date;
7. compare amounts located at different dates;
8. add amounts situated at the same date;
9. add amounts situated at different dates;
10. choose a reference date;
11. move an amount from one date to another;
12. collapse a series of periodic amounts into one single amount;
13. explode an amount into a series of periodic amounts.

Similarly, in the subset of **mechanical physics** refered to above, some activities would be:

14. compute a torque;
15. compute an angular acceleration;
16. compute a moment of inertia.

In many cases, a PS activity can be rephrased into, restated as, a different one, of a lower *abstraction level*, because more immediate, more down-to-earth, closer to the problem to be solved. For example, in **mechanical physics**, assuming that the torque and the moment of inertia of a given solid body are known (either given or previously computed), the activity "compute the angular acceleration" (activity 15) would be expressed as, or translated into "apply formula (7)", an instance of the lower-level activity 3. A PS physics tutor is presented in [Gertner & VanLehn, 2000].

Sometimes, a PS task may also be divided into smaller ones, letting us use again the notion of *complexity levels* in these tasks. For example, in **cost engineering**, comparing two amounts situated at different dates implies:

- first, choosing a reference date at which to make the comparison;
- then, moving either (or both) amount(s) from its (their) present date(s) to the reference date;
- finally, comparing the amounts, now both located at the same reference date.

These subactivities (of types 10, 11, and 6 respectively in the sample list above) thus appear to be of a lower complexity level than the initial one (of type 7). However, it is interesting to note that, although activity 7 turns out to be more complex than activity 6 (the latter is part of the former), both are stated using the same *abstraction level*.

It may also happen that some lower-level activities can *only* appear as components of a higher-level one. For example, still in **cost engineering**, the activity "drawing a temporal diagram" (type 5 above) implies the following tasks, which can only be accomplished as part of that activity (hence their identification in this paper from 5a to 5d):

5a. draw a timeline to encompass all periods implied by the problem data;
5b. draw arrows representing the amounts involved in the problem data;
5c. if necessary, split an amount (or each amount in a series) to simplify the computations;
5d. qualitatively draw a special arrow to represent the expected result of the computations to be made.

In that case, activity 5 is both of a higher complexity level and of a higher abstraction level than any of its subactivities.

4 Tutoring knowledge

We now briefly present the *tutoring knowledge* (TK) in order to help the reader to better apprehend the relationships of that knowledge with DK and PSK. This third type of knowledge contains all tutoring processes enclosed in the ITS. It is not directly related to the teaching domain or to problem solving, but is there to help the student understand, assimilate, and master the knowledge included in DK and PSK [Gagné & al., 1992; Gagné & Trudel, 1996].

The main system activities using TK are:

- ordering and formatting the topics to be presented to the student;
- monitoring a tutoring session, i.e., triggering the various tutoring processes according to the system tutoring goal and the student's actions; such monitoring may imply giving explanations, asking questions, changing to another type of interaction, etc.;
- in a PS-domain, while the student is solving an exercise, monitoring the student's PS activities: understanding and assessing these activities, giving advice to correct or optimize them, giving hints or partly solving the exercise at hand (as required by the student or by the tutoring module), etc.;
- continuously analysing the student's progress in order to improve the efficiency of the tutoring process.

The advantage of separating the tutoring knowledge from the knowledge of the domain to be taught has been emphasized long ago [Goldstein, 1977; Sleeman & Brown, 1982; Clancey, 1986; Wenger, 1987], and lies in the reusability of TK in various domains. In the case of PS-domains, the domain to be taught clearly encompasses both DK and PSK; indeed, the term "domain knowledge" applies to DK if referring to the knowledge type, and to DK + PSK if referring to the knowledge to be acquired. Therefore, as shown in the introduction, in a PS-ITS, knowledge ends up being separated into three categories rather than two.

We believe that the tutoring processes are triggered by *tutoring goals* which depend on the current educational setting and learning context. The role of tutoring goals has been discussed in several works, some of the most recent ones dealing with task and instruction ontologies [Mizoguchi, 1999]. In the current state of our research, our assumption is the following: the underlying hierarchy or hierarchies governing the way tutoring processes interact with one another is not related to these processes *per se*, but rather to the *current goal* to be attained when they are invoked. The current goal varies during the session, depending on the student's actions or difficulties, following a dynamically built abstraction-based hierarchy. If our assumption turns out to hold, then the *dynamic structure of educational goals and subgoals* — which itself depends on the student's desires or abilities, the main underlying objective of the tutoring system, the student's state (e.g. of tiredness, etc.) and performance, etc. — will determine the succession of tutoring processes activated and tutoring interactions taking place. To our knowledge, the use of abstraction levels to induce a dynamic hierarchy of tutoring goals is new, as is the assumption that such a hierarchy will play a major role in activating the various tutoring processes and student–system interactions. Learning goals have been used by Towle [2000], but only for educational simulations, not for tutoring processes in general.

5 Educational interests of abstraction and complexity levels

In the above sections, we have sketched a complexity– and abstraction–level approach to help model the three types of knowledge involved in a PS-ITS. In this section, after clarifying these notions in section 5.1, we present the educational interests of our model. Sections 5.2 to 5.4 focus on the type of knowledge respectively presented in sections 2 to 4. Section 5.5 summarizes that discussion with some overall pedagogical interests of our approach.

5.1 An informal definition of abstraction and complexity levels

In the first three sections, we only refered to abstraction and complexity levels. Here, we try to define these notions better and in a more generally applicable way. Both notions are based on the common notion of *refinement*, but differ in how the refinement is made: in a general way, abstraction is based on, or refers to, *expressiveness* or *scope*, whereas complexity is based on, or refers to, the *number of components*.

For *concepts*, taking geometry as an example, a polygon has a higher abstraction level than a triangle or a square, because the number of sides in a polygon is indefinite, but a lower abstraction level than a set of segments, because these segments in a polygon are forced to be pairwise adjacent; a square has a higher complexity-level than a triangle, because it has more sides, and also because there are constraints (re. size and angles) between these sides. In cost engineering, we saw that the factors Φ_{FP} and Φ_{AP} are expressed at the same abstraction level, although Φ_{AP} has a higher complexity level, because of the way it is defined and computed. A similar distinction between abstraction levels and complexity levels holds for the *relations* they express.

For *problem-solving activities*, we have similar distinctions, as shown in section 3.2 with several examples.

Finally, the same holds for *tutoring processes*, or student–system interactions. For instance, the ITS task of tutoring a student while s/he is solving a problem will turn out to be of a higher complexity level if the student encounters more difficulties, although the abstraction level of this process does not depend on the particular student being tutored or on the particular problem being solved. On the other hand, reacting to a student request for hint, or for explanation, is of a lower abstraction level than the previous one; however, there again, the complexity of that task will depend on the specific student request (some simply formulated questions may have quite complex answers!), and will eventually depend also on the way the student is or is not satisfied with the initial system response.

Such level-based distinctions have also been made, for example, by Mizoguchi [1999]. Note that, although the statement "A has a higher abstraction level than B" is clear and may be true, we think that the number of abstraction levels between A and B is not defined, because that number would depend on the modelling effected; for the same reason, it would be even more meaningless to try to assign a numeric value to these levels.

5.2 Domain modelling

The definition of concepts from the simplest to the most complex induces a long-time known *presentation order* for the subject matter. Similarly and in addition, the factor hierarchy described in section 2 for cost engineering lets us derive an *order for the presentation of factors* to the student, from the lowest (simplest) level up to the highest, i.e. with increasing understanding complexity. That does not imply that such an order is unique, or even the best (e.g. a student's personal interests might make another order more motivating for him/her), but it is justified by our model. This presentation order may itself induce, like for domain concepts, a possible *order for prerequisites*; e.g., if a student experiments difficulties to deal with Φ_{AP}, has s/he well mastered Φ_{FP}, a conceptually simpler factor?

Moreover, the factor-induced intermediate abstraction levels will permit the ITS to exhibit a *sharper modelling of conceptual errors*. For example, the source of an understanding error concerning one of the two relations in equation (2) or (3) or (7) (see also figure 1) is much easier to identify using the corresponding factor, either as a definition error or as a usage error, than an error concerning the global equation (1), where the definition and application relationships are not made explicit, and therefore are impossible to distinguish. Similarly, an error using a Φ_{AP} factor may be diagnosed as possibly resulting from an insufficient mastery of the simpler factor Φ_{FP} as concept (which in turn will be diagnosed as related either to its definition, or to its usage). Similarly in physics, if the student stumbles on concepts like angular acceleration or moment of inertia, has s/he mastered the simpler although similar concepts of acceleration or mass?

Abstraction and complexity levels on domain elements (concepts and relations, possibly including factors) can then be used to introduce various abstraction levels of explanations. Such explanations can then be tailored to the student's questions, and adapted to the reminders possibly needed by the student.

5.3 Problem-solving modelling

The problem-solving activities briefly presented in section 3 naturally display abstraction and complexity levels. Indeed, a standard problem can usually be divided, possibly in more than one way, into major steps, which can then be split into simpler substeps. As explained in 5.1, each subactivity in that case may be either simpler (lower complexity level) or more concrete (lower abstraction level) than the original one, or both.

In a first development stage, these abstraction and complexity hierarchies, both for domain elements and for problems to be solved, can ease the *definition of exercise types* to be implemented into the ITS, and can ease the tutor module task of *choosing the exercise type* to challenge the student with. Later, once that basic system is operational, the same hierarchies can help develop an automatic *exercise generator* dealing with the domain elements to be mastered by the student. That approach will then help the student to acquire a better critical mind about the relative importance of *problem–solving* knowledge vs. *domain* knowledge.

As for domain elements, abstraction and complexity levels can be used to introduce various types of *explanations* about the problem to be solved, varying both in abstraction (focus level, terms used, references made) and in complexity (quantity of details, possible references to the problem substeps). Moreover, our approach will lead the student to focus specifically on the *activities* for which s/he needs more tutoring, with the abstraction and complexity levels appropriate to his/her individual case.

5.4 Tutoring modelling

Functioning mode		Domain-presentation mode	Demonstration mode	Domain-assessment mode	Exercising mode
Main type of knowledge involved		Domain knowledge	Problem-solving knowledge	Domain knowledge	Problem-solving knowledge
Student's main goal		To learn (acquire or improve knowledge)		To assess his/her learning	
Direction of the knowledge transfer		System → Student		Student → System	
Typical interaction	Trigger (start)	The student asks the system...		The system prompts the student...	
		some information about a domain theoretical element	to solve a practical problem or to coach him/her in problem solving	to develop a domain element	to solve a practical problem
	Knowledge exchange	The system presents...		The student presents his/her view of...	
		the requested element	a possible solution to the requested problem	the requested element	a possible solution to the given problem
	Result (closure)	The student expresses his/her understanding...		The system assesses the student's answers, and possibly corrects them	
		of the element	of the problem solution		

Table 2 — Characteristics of the four typical operating modes of a problem-solving ITS.

As presented in section 3.1, the distinction between DK and PSK leads to the natural definition of four operating modes. Their main characteristics are recalled in Table 2.

The successive tutoring goals aimed at by the system (see section 4) are likely to result in a chain of recursive calls of the tutoring processes invoked. This recursivity will or will not be direct, depending on the tutoring interaction types being chained: the system might decide to temporarily change between interaction types, e.g. to respond to the student's actions or requests. However, the potential length of this chain is only apparent: because of the abstraction hierarchy of tutoring goals, each newly invoked process will be called with a *narrower scope* and/or a *lower complexity*, which eliminates the risk of "forgetting" the initial tutoring goal or of running into an infinite loop.

More generally, tutoring the student may take the form of explanations, guidance, hinting, or even partially solving the exercise on which the student is currently working. The level at which these will be conducted will depend on the current tutoring goal (see section 4). We think our approach is close to that of VanLehn and his colleagues [2000], although they focused their attention on fading and deepening (a particular result of the tutoring interactions) rather than on the current pedagogical goal (the cause for these interactions).

5.5 Overall interests of these abstraction and complexity levels

Abstraction levels are certainly not new. What we think is new is to use them in a systematic way to shed a uniformizing light on the ITS design and operation, and to make it more user-friendly once implemented.

First, they may help to give a better tailoring to the system tutorial interventions to fulfil the student's needs and the system tutoring goals, thus improving its conviviality and efficiency.

Then, all the capabilities presented above should result in smoother, more "natural", human-like interactions with the student. This improved ability to reproduce a human teacher's behaviour contributes again to make the system more user-friendly, and more likely to be used by the student.

Finally, although that aspect is not in the scope of this paper, our refinement of the three types of knowledge as described in sections 2 to 4 paves the way to the conception and the implementation of a structured error model, and eventually of a structured student model.

6 Conclusion

This presentation of a possible knowledge structure for PS-domains, which emphasizes the separation between domain knowledge and problem-solving knowledge, shows how a general functioning theory of such an ITS — namely the four operating modes described in sections 3.1 and 5. — can naturally be derived.

Moreover, the abstraction and complexity levels highlighted throughout this paper can be used as a common guideline to help finding an appropriate representation for each one of the three knowledge type, and thus can help creating more efficient ITSs. More generally, this guideline can shed a uniformizing light on the system design, although it has never been used in a systematic way in the design or implementation of an ITS.

We thus hope to bring some contribution to the general and important problem of finding a generic architecture for intelligent tutoring systems.

References

[1] Anderson John R.: *Cognitive Modelling and Intelligent Tutoring*. Washington, DC: National Science Foundation (1986).

[2] Anderson John R. & B. J. Reiser: "The LISP Tutor". *Byte* **10**, no. 4, p. 159-175 (1985).

[3] Brodie Michael L., John Mylopoulos & Joachim W. Schmidt, eds.: *On Conceptual Modelling, Perspectives from Artificial Intelligence, Databases, and Programming Languages*. New York: Springer Verlag (1984).

[4] Brown John S. & Richard R. Burton: "Diagnostic models for procedural bugs in mathematical skills". *Cognitive Science* **2**, p. 155-192 (1978).

[5] Clancey W. J.: "Qualitative student models". *Annual Review of Computer Science* **1**, p. 381-450 (1986).

[6] Clocksin William F. & Christopher S. Mellish: *Programming in Prolog*. Berlin: Springer-Verlag (1981).

[7] Findler Nicholas V., ed.: *Associative Networks*. Orlando, FL: Academic Press (1979).

[8] Frasson Claude, Gilles Gauthier & Alan Lesgold, eds.: *Intelligent Tutoring Systems*, Proc. of the *3rd International Conference, ITS'96*, Montréal (Canada), 12-14 June 1996. LNCS 1086, Berlin: Springer (1996).

[9] Gagné Denis & André Trudel: "A highly flexible student-driven architecture for computer-based instruction". [Frasson & al., 1996], p. 66-74 (1996).

[10] Gagné Robert M., Leslie J. Briggs & Walter W. Wager: *Principles of Instructional Design*, fourth edition (1st edition published in 1974). Orlando, FL: Harcourt Brace Jovanovich (1992).

[11] Ganeshan R., W.L. Johnson, E. Shaw & B.P. Wood: "Tutoring diagnostic problem solving". In [Gauthier & al., 2000], p. 33-42 (2000).

[12] Gertner A.S. & K. VanLehn: "ANDES: a coached problem-solving environment for physics". In [Gauthier & al., 2000], p. 133-142 (2000).

[13] Goldstein Ira P.: *The Computer as Coach: an Athletic Paradigm for Intellectual Education*. AI Memo 389, AI Laboratory. Boston, MA: Massachusetts Institute of Technology (1977).

[14] Goldstein Ira P.: "The genetic epistemology of rule systems". *International Journal of Man-Machine Studies*, Vol. 1, No. 1, p. 51-77 (1979).

[15] Gauthier Gilles, Claude Frasson & Kurt VanLehn, eds.: *Intelligent Tutoring Systems*, Proc. of the *5th International Conference*, Montréal (Canada), 19-23 June 2000. LNCS 1839, Berlin: Springer (2000).

[16] Kim Won & Frederick H. Lochovsky, eds.: *Object-Oriented Concepts, Databases, and Applications*. ACM Press, Reading, MA: Addison-Wesley Publ. (1989).

[17] Kowalski R.: *Logic for Problem Solving*. Berlin: North-Holland (1979).

[18] Lelouche Ruddy & Jean-François Morin: "Introduction of pedagogical concepts in domain modelling for an ITS". *European Journal of Engineering Education*, Vol. 23, No. 2, p. 255-271 (June 1998).

[19] Lelouche R. & J.-F. Morin: "Knowledge types and tutoring interactions in an ITS in a problem-solving domain". *FLAIRS 1997, Proc. of the 10th International FLorida Artificial Intelligence Research Symposium*, Daytona Beach (Florida, U.S.A.), 10-14 May 1997, p. 62-66 (1997).

[20] Lenat D., F. Hayes-Roth & P. Klahr: *Cognitive Economy*. Working Paper HPP-79-15, Stanford Heuristic Programming Project, 46 pages. Stanford, CA: Stanford University (June 1979).

[21] Malec J.: " Knowledge elicitation during dynamic scene description ". *ACM–SIGART Newsletter: special issue on knowledge acquisition*, No. 108, p. 162-163 (April 1989).

[22] Mizoguchi Riichiro: "Ontology-awareness for intelligent instructional systems". Invited talk. *Advanced Research in Computers and Communications in Education, Volume 1* (G. Cumming, T. Okamoto & L. Gomez, eds.), Proceedings of the *7th International Conference on Computers in Education, ICCE 99*, Chiba (Japan), 4-7 November 1999. Amsterdam, Netherlands: IOS Press, p. 45-52 (1999).

[23] Morin J.-F.: *Conception of an intelligent tutoring system in cost engineering: knowledge representation, pedagogical interactions, and system operation*. Master Thesis, Département d'Informatique. Québec, Canada: Université Laval (1998).

[24] Patel Ashok & Kinshuk: "Intelligent tutoring tools in a computer-integrated learning environment for introductory numeric disciplines". *Innovations in Education and Training International*, Vol. 34, No. 3, p. 200-207 (1997).

[25] Sleeman D. H. & John Seely Brown, eds. : *Intelligent Tutoring Systems*. London: Academic Press (1982).

[26] Sowa John F.: *Conceptual Structures*. Reading, MA: Addison-Wesley (1984).

[27] Towle B.: "Using student task and learning goals to drive the construction of an authoring tool for educational simulations". In [Gauthier & al., 2000], p. 173-181 (2000).

[28] VanLehn K., R. Freedman, P. Jordan, C. Murray, R. Osan, M. Ringenberg, C. Rosé, K. Schulze, R. Shelby, D. Treacy, A. Weinstein & M. Wintersgill: "Fading and deepening: the next steps for ANDES and other model-tracing tutors". In [Gauthier & al., 2000], p. 474-483 (2000).

[29] Wenger, Étienne: *Artificial Intelligence and Tutoring Systems*. Los Altos, CA: Morgan Kaufmann (1987).

Using Highly Sophisticated Middleware For Building Arbitrarily Distributed Teaching Environments

Christof Dallermassl, Heimo Haub, Harald Krottmaier,
Klaus Schmaranz, Philipp Zambelli
{cdaller,hhaub,hkrott,kschmar,pzamb}@iicm.edu

Institute for Information Processing and
Computer Supported new Media

This paper deals with the development of highly sophisticated teaching environments. We took a look at the requirements that such a system has to fulfill to meet the needs of our modern society and to remain easily adaptable to forthcoming, new technologies. The results of our research show that the cost for design and implementation of a distributed teaching environment can be dramatically reduced by using a middleware system. We thereto present the concepts of the Dinopolis middleware system which is highly modular and extendable and show how it may be used as the basis for the development of teaching environments. Using the Dinopolis middleware system not only eases the process of development but also guarantees that new technologies are becoming available without any further effort.

Keywords: **Distributed Teaching Environment, Middleware, Application Framework, Distributed Object System**

1 Motivation

We are living in times of ongoing rapid changes. New technologies support our everyday life in a way we could not even imagine a few years ago. New media are also disrupting our old fashioned way of thinking about education and pave the way for new visions [5].

Many different approaches for developing electronic teaching environments already exist. All of them have in common that they try to react on the changes taking place in our modern society. Acquired knowledge is becoming obsolete in a short amount of time and has to be updated and expanded. The term "Life-long Learning" is becoming more and more important. Teachers and students are often no longer present at the same location. They are not even active at the same time. Therefore lectures have to be available whenever and wherever one likes. On the other hand new technologies have to be made available as soon as possible. As an example consider the new WAP protocol which could make it possible to pursue a course using a mobile phone. This implies a highly extensible and expandable system.

Teachers preparing a course often want to use already existing data. This data might be distributed among the network and might only be available in proprietary formats. It is an unbearable effort to collect and extract this data to feed it into a teaching environment. It has to be possible to easily include existing data with no further effort. It has to be possible to add or manipulatable information at any time without knowing any details of the underlying systems. When updated information has become available, documents have to be replaced. Nevertheless replaced information should not be lost as it shows the history of a course and might be interesting for some research. Thus a version control mechanism is desired as well.

Students need easy access to the teaching system, no matter which system they use (platform, browser, etc.). The way in which information is provided has to be adapted to the user's special needs and interests. The users' skills have to be taken into account as well as the capabilities of the system used. As an example, consider the network bandwidth with which the user is connected. Lower bandwidth could be considered by sending images of a lower resolution and just sound without the video. Users of different skills need information prepared differently, which means that different applications are provided for novice and expert users. Thus the system has to be highly customizable, not only by the user but also automatically by the system itself. Different views at the information space are desirable as well. As an example the information could be provided in different languages or with different, localized examples.

Another important point is the use of background libraries [7] to clarify terms. This includes dictionaries, encyclopedias and glossaries which again have to be adopted to the user's needs. As an example native English speakers would need an explanation of an unknown English term in English, whereas German native speakers would like to get a German explanation.

Though students may be distributed among the world they need the ability to discuss certain topics with the teacher or among themselves. Collaboration and communication tools are therefore required. This includes chat, discussion forums, video conferencing tools, etc.. Another important aspect is the possibility to add annotations and make personal notes to a topic which may only be seen by a specified group of others, maybe even excluding the teacher. This directly leads us to another interesting aspect in a teaching environment: user and group access management. Information presented in a course as well as additional information has to be protected from unauthorized access. The more sophisticated the user management is, the more configurable the system becomes.

Obviously the best user access management system is not worth anything if the transmission via the network is not secure at all. Highly sensible data has to be additionally encrypted. Imagine companies teaching their personnel internal knowledge.

For performance reasons hybrid systems are often desirable. Most of the courseware comes along on a CD-ROM. Only the logical part of a course could reside on a server. Thus lessons can be arbitrarily constructed using the information stored on the CD-ROM. Only additional or updated information has to be fetched via the net. This keeps network traffic low without omitting special kinds of media such as videos. Hybrid systems are also kept up-to-date more easily, since outdated information stored on the CD-ROM can be replaced dynamically by new data from the server.

Linking is a good way to increase the value of information if used for defining the course flow [6]. This makes it easier to add or replace information used in a course. That way even statically stored information (e.g. on a CD-ROM) can be dynamically restructured to meet different needs. Since links are often subject to change or become invalid (especially links to external resources), a highly sophisticated link management is required to keep the system consistent.

Developing a teaching environment that meets all the functionality described above implies a big effort for implementing the base functionality. Most of this functionality has little to do with a certain teaching concept pursued. It requires a lot of time to develop the system so far that the actual vision can be put into practice. This paper describes how the cost for developing can be reduced dramatically when using a middleware system. This offers more time to concentrate on the crucial points of new ideas. In the following sections we present the abilities of the Dinopolis [3] middleware system which is highly integrative, expandable and configurable and provides an easy-to-use interface for building distributed applications.

2 A Distributed Environment

As already mentioned, distance learning gets more important every day. Teacher and students do not have to be physically present in a classroom. The teacher for example holds online lessons in his office, while the students follow the lessons from at home or from some Internet terminal on the campus. It is important to recognize that the students do not only need to follow the lessons conducted by the teacher, but also need the possibility to interact or communicate with the teacher during and after the lessons.

Please note that not only human beings are separated in such a distributed environment but also resources can be spread across the network. As an example, the lecture notes can be stored on the teacher's laptop while a chat tool may be located on the university server.

The main task of modern teaching environments is to bridge this physical separation of teachers, students and resources. This can be supported by using the Dinopolis middleware system. A middleware system adds another layer to the teaching environment. It decouples the network and data form the actual teaching application. It somehow resides between them. The Dinopolis middleware system allows the integration of arbitrary systems and provides a uniform way of access. Applications using Dinopolis do not have to worry about where the data is located and which protocol has to be spoken to retrieve the data.

From an application's point of view there is no difference if the information resides on a web server, on a database server or just on the local file system. The underlying systems may even be exchanged transparently. An important point is that integration within the Dinopolis system is not done on a common denominator basis but on the contrary, additional functionality is added when needed. As an example a file system can be combined with a database system to allow the storage of meta data which the file-system itself does not support.

The most intuitive way to access data on a remote Dinopolis system is by "merging" the two Dinopolis systems, since it allows transparent access to remote Dinopolis instances. In other words the middleware layer bridges the physical separation and applications can be written as if they would run on one single system. Nevertheless the Dinopolis *External Access Gateway* concept allows access in various forms. At the moment this includes HTTP, FTP, WAP, LDAP, CORBA, RMI, etc.. CORBA and RMI are remote object systems which enable the use of objects over the network. In the case of CORBA [8] the Dinopolis object system can be used by programs written in various programming languages.

Thus no matter how distributed teachers, students and resources are, the Dinopolis middleware system acts as if they were all local. For a more sophisticated discussion of middleware systems and Dinopolis see [2].

3 Data And Course Flexibility

As already mentioned information is subject to change and the amount of information grows dramatically nowadays. Data must not be stored within a static teaching environment providing no possibility to adapt the data. It is also desirable to change the course flow dynamically to increase the value of information. As an example it has to be possible to add actual information to a course at any time. Thus developing a teaching environment built on static information is not worth the effort.

The Dinopolis middleware system allows to add, exchange and modify data at any time without needing to know any specific details about the underlying heterogenous information space. Version controlling guarantees that no information is lost. The distributed concept of Dinopolis allows to store data at arbitrary places transparently. Applications do not have to worry about that. This makes it relatively easy to develop so called hybrid systems, which combine local statically stored data (e.g. on a CD-ROM) with dynamically retrieved data (e.g. from the web). Due to the network bottleneck hybrid systems have the advantage that they do not have to omit bandwidth consuming information, such as videos, since they can be retrieved locally. Only updated information has to be downloaded via the net.

The course flow is best modeled using links. This makes it possible to insert or remove any kind of information without having to construct a complete new course. The problem with links is that they often become invalid or point to unintentionally changed data. Dinopolis comes along with a fully featured consistent link management system. Links do not point to a location but to an object. Thus even if objects are moved in the Dinopolis system, links remain valid. It is also possible to add meta data to links to provide additional information. Again the possibility to set links does not depend on the abilities of the underlying systems. As an example imagine a courseware CD-ROM. Since a CD-ROM is read-only it is impossible to add or modify links directly on that medium. Nevertheless Dinopolis adds this functionality using an external link database (e.g. Oracle). This database can easily be exchanged transparently which then leads to a completely new course or course flow.

Dinopolis is able to handle arbitrary document formats. This is achieved through the Dinopolis internal document model, which follows the *Document Object Model* (DOM) specification of the W3C [9]. DOM is a highly accepted standard which supports the exchange of data between various kinds of different document formats. The internal document model again makes the whole system independent from the underlying ones.

4 An Extendable And Exchangeable Environment

A modern teaching environment not only has to be flexible concerning the course data and course flow. As applications evolve new technologies and requirements arise which require modifications of existing teaching environments.

Dinopolis is built on a completely modular basis. This allows to add or replace arbitrary parts of the system without producing unexpected side effects on the remaining parts of the system. Since Dinopolis is completely written in *Java*, it is possible to load new modules via the network even at runtime. Statically designed systems would not only have to be completely rewritten but also have to be redistributed among all users. As an example for a newly available technology consider the WAP protocol which could make it possible to pursue a course using a mobile phone. Dinopolis only requires a small WAP speaking module to be added and all applications using Dinopolis are becoming WAP aware. This means new technologies are available without any further effort. By the way adding new functionality to an application by hand would also require to be familiar with all the underlying details, which might be unnecessarily time consuming. Since Dinopolis is an *Open Source* project it is also guaranteed that modules supporting new technologies will be rapidly available.

Dinopolis is not only modular concerning external communication gateways. Internal parts of the Dinopolis system may be exchanged or added transparently as well to meet different needs. As an example let's take a look at the *User Access Management* and *Security* system which will be explained in section 6 more in detail. For some systems it is sufficient to define simple read and write access rights for users. Other systems require a much more sophisticated mechanism, such as a rule based one which only grants access if certain complex conditions are fulfilled. As an example students who have already passed an exam might get access to the results which are protected otherwise.

Teaching environments transmitting highly sensible data might also require to encrypt the information sent over the network, while a public system would not require it. This implies a *Security Manager* that can easily be adapted to different strategies. This also includes that some systems demand a user authentication based on smart-cards, finger prints or retina scans.

Dinopolis makes it easy to exchange or adapt internal parts to meet different requirements without the need to write a new application or modify existing ones. Dinopolis even allows to exchange the internal data structure or communication protocol used. At the moment data is stored according to the XML standard [1] and communication between Dinopolis instances is done using RMI or CORBA. As soon as new technologies and standards become available they will be integrated as well.

5 A Highly Customizable Environment

In a modern teaching environment we expect the main parts of the system to be highly customizable. The whole system has to allow users to turn on and off certain features according to their needs and their systems' capabilities. On the other hand applications and tools have to be customized depending on personal settings. Additional communication tools have to be provided to support a better information flow between users. The Dinopolis system is highly modular which allows adding and removing of integral parts, features, and services at runtime.

The users should have the possibility to choose between a variety of tools and applications that support their studies, depending on the users' skills and the used network connection.

The Dinopolis system makes it easy to write distributed applications. Existing tools can be reused and adapted for certain lectures as needed. It is easy to configure the system with different applications at runtime and the Java Classloader allows starting applications which reside on the local terminal, a CD-ROM, or to download them from an application server.

It is important that only those applications are part of a certain course, which are actually necessary.

Next it should be possible to customize the applications and tools themselves. As an example consider a video-conferencing tool which can be run with different frame-rates and resolutions. On the other hand it has to be possible to turn on and off special features like, for example, strong data encryption. Applications written for Dinopolis can make use of certain features or not, according to the environment in which they are used.

Apart from the applications and tools used, such a teaching environment has to allow customizing the users' view on the data stored in the information system. As an example consider a teacher who wants to adapt the data representation according to the different skills and needs of the users.

Therefore the Dinopolis system provides so-called Views, which support different data-representations. Using a highly sophisticated link management (see section 3) the data representation can be customized in a very high degree.

Another considerable point is that it has to be possible to decide where the data is actually stored. So it could be desired to store certain data on a file-system on the users' terminal or in a central database. It should be possible to transfer these data from one storage medium to another. Since the Dinopolis system uses an internal data representation which is independent from the medium it is stored on, it is possible to store the data on different devices according to the actual configuration of the system.

6 User Access Management And Security

All distributed application have in common that their reliability and consistency heavily depends on the security and user access management. Unauthorized access to certain resources has to be prevented. This is also appropriate for a teaching environment. Just imagine students modifying and corrupting course data or exam results.

There exist many different approaches to solve this security task. Which one is appropriate for a certain environment depends on the desired degree of security. As already described in section 4 a simple security system differentiating between read and write access rights may be sufficient for some applications. More sophisticated environments could require a rule based access control system or secure (encrypted) transmission over the network.

One of the main parameters to measure well designed software is its level of reusability. In this case this means that security strategies have to be exchangeable easily. As already mentioned replacing the Dinopolis security concept is no complex task due to its modular structure.

Another point of high interest concerning access management in a teaching environment lies in the fact that data is distributed among heterogenous systems, all of which coming along with different or even no security management. It has to be guaranteed that users are forbidden to access data which they would not be allowed to access otherwise. Additionally if a system does not support any access control (e.g. MSDOS file-system) it has to be possible to add a security functionality. Dinopolis allows to integrate existing security mechanisms transparently. It is also possible to add access control to systems which do not support that by themselves. Thereto Dinopolis uses its integrated link management system. Access rights and rules are simply assigned to users and data through links which may be stored in any arbitrary external database. This concept makes it even possible to assign access rights to read-only systems (e.g. CD-ROM). Last but not least Dinopolis supports mandatory as well as discretionary access control. This means that access rights may be assigned to users as well as to data, which allows highly sophisticated combinations of rights.

For a more in detail discussion of access management in distributed systems and in Dinopolis see [4].

7 Conclusion

The Dinopolis open source middleware system presented in this paper is a powerful tool for building arbitrarily distributed teaching environments. It allows to integrate data from heterogenous information space transparently and makes them available through a common interface. Thereby functionality is not reduced to a minimum but on the contrary, additional functionality is appended where needed.

The modular concept of Dinopolis makes it easy to adapt the system to specific needs without producing unexpected side effects on the remaining parts of the system. Its high customizability makes it possible to configure the system to meet personnel needs. Additional communication tools such as chat, video conferencing, etc. may be used together with the teaching software without any further effort.

Courses may be constructed using all available distributed resources which also eases the creation of hybrid systems. Easy access to the system is possible not depending on the clients' platform or desired protocol.

The conglomerate of Dinopolis' features presented allows developers of teaching environments to concentrate solely on the crucial points of their ideas. Thus we believe that Dinopolis is going to play an important role in the future development of distributed teaching environments.

References

[1] Bray T. et. al.: "Extensible Markup Language (XML) 1.0 (1998)", available online at `http://www.w3.org/TR/1998/REC-xml-19980210`.

[2] Dallermassl, Ch.: "Aspects of Integration of Heterogenous Server Systems in Intranets - The Java Approach", IICM (Institute for Information Processing and Computer Supported New Media) (1999)

[3] Dinopolis: "Homepage of the Dinopolis Open Source Project", available online at `http://www.dinopolis.org`

[4] Haub, H.: "Aspects of Access Management in Heterogenous Distributed Object Systems", IICM (Institute for Information Processing and Computer Supported New Media) (1999)

[5] Grudin, J.: "Groupware and Social Dynamics: Eight Challenges for Developers (1994)".

[6] Van Dyke, J.R. & Sollins, K. R.: "Linking in a Global Information System, World Wide Web Journal - A Publication of the W3C" (1995), available online at `http://ana-www.lcs.mit.edu/anaweb/pdf-papers/tr-659.pdf`.

[7] Marchionini, G. & Maurer, H.: "Digital Libraries as Components of Modern Computer Supported Learning Environments" in proceedings of ED-MEDIA 95. World Conference on Educational Multimedia and Hypermedia, Graz Austria 1995.

[8] Object Management Group, Inc.: "CORBA Language Specifications Summary", OMG (2000) available online at `http://www.omg.org/store/publications.html`.

[9] W3C: "Document Object Model (DOM) Level 1 Specification" (1998), available online `http://www.w3.org/TR/1998/REC-DOM-Level-1-19981001`

Xtrain: A GUI based tool for Multimedia Presentations, Instruction, and Research

Xiagen Hu* & Scotty D. Craig**
The University of Memphis
University of Memphis, Department of Psychology, Memphis, TN 38152-6400
(901) 678-3608
*E-mail: *xhu@memphis.edu, **scraig@memphis.edu*

Xtrain is a program for scripting and presenting multimedia displays. This program was developed in the Advanced Learning Technologies Laboratory at the University of Memphis and has been used in variety of psychological experiments. This program can combine such multimedia formats as Microsoft agent, Macromedia flash, director and many others that are available for scripting under a GUI Windows environment. Furthermore, Xtrain offers a variety of options for testing styles.

KEYWORDS: **Conversational agents, multimedia applications, Xtrain, Microsoft Agent**

1 Introduction

The idea of embodied conversational agents has generated considerable interest in the realm of HCI recently. Unfortunately, for the most part this has been metaphorical, because computers could not support the needed software. In order for the computer to fully support embodied conversational agents, they would need software that could produce and control many human-like characteristics, such as conversational behaviors, with the ability to mediate the flow of conversation by the use of such things as facial expressions, hand movements and voice intonations [1].

Microsoft made one such attempt at this with their Microsoft Agent program. Microsoft Agent is an interactive interface with conversational capabilities that are embodied in an animated character agent. One example of this is the helper agent in newer versions of Microsoft products such as Microsoft Word and Microsoft PowerPoint. Microsoft Agent is also an optional program for windows 9x operating systems and is available for free download at the Microsoft site. It is compatible with all MS Windows platforms starting with Windows 95. Thus, this agent is readily available for widespread use [5]. The purpose off this paper is not to review Animated Agents for a review, see Johnson, Rickel, and Lester [4].

Furthermore, recent research has shown that the correct use of multimedia presentations can enhance the learning and memory from presented materials. Multimedia in this case refers to any type of pictorial information presented with textual information. However, this form of learning works best with pictorial information shown as an animation that is then coordinated with a narration of any textual information that would be needed [6]. Under the cognitive theory of Multimedia learning, there are three main rules that should be considered for scripting of a multimedia presentation: Spatial contiguity, Temporal contiguity, and Modality. The spatial contiguity effect states that relevant and related concepts should be presented in the same general area of each other (e.g. labeled words should be closer to the object they label than other objects on the screen). The temporal contiguity effect informs us that the various forms of media used during a presentation should correspond with each other by occurring at the same time. Lastly, the modality effect says that if two types of information are presented in the same type of modes, it will hinder learning. However, this can be overcome by presenting information in two modalities. So, printed text and animation on a computer screen would be a hindrance to learning, but a narration and an animation would not [7; 8].

Since Microsoft launched the first version of Microsoft Agent, users and developers have provided a lot of resources for use with the program (e.g., some information can be obtained

BEST COPY AVAILABLE

fromhttp://www.msagentring.org/agentringmain.html). There are several innovated approached both in the use and the scripting of Agents. For example, mash.exe (http://www.bellcraft.com/mash/) provide a very useful scripting tool for agent programming. Many of these programs have been examined, including Mash, and while they have the ability to control Microsoft Agent, they are lacking the ability to synchronize the Agent program with other forms of multimedia.

The Advanced Learning Technologies laboratory at the University of Memphis developed Xtrain as a way to incorporate embodied agents (Microsoft Agent) and other forms of multimedia into instruction, research, and presentations. Psychologists have used products such as Mel©and Super lab©to run experiments, but these programs cannot incorporate newer technology. Xtrain provides ways to script many different kinds of presentations, including Microsoft Agent, audio and video clips, HTML, Macromedia flash files, Macromedia Director files, and many graphics file formats [2, 3].

This software program serves a duel purpose. It is both an authoring tool and presentation tool. These work together to form a powerful and versatile tool for the presentation of various multimedia displays as well as data collection.

2 Authoring tool

The authoring side of the program has two levels (a) overall organization of frames and (b) detailed construction of individual frames. The overall methodology is similar to the SuperLab program used in experimental psychology. The Presentation is organized in terms of a tree structure with each node in the tree as a pointer to presentation frames. Each frame consists of the smallest unit of information and the frames are logically contingent upon each other. Such tree structure serves as basic navigation guidelines. However, the navigation path can be quite flexible depending on the needs of the user. The tree structure can be created using a user friendly GUI. Each frame corresponding to the tree nods can be any of several formats such as text art, pictures with hotspots, video/audio clips, agent interactions, and animations.

Xtrain has extensive options for frame editing. The program has been arranged so that the different editing functions displayed as individual property tabs. Each tab corresponds to a specific multimedia format. A description of the property tabs will follow.

Property Tabs

2.1 Frame Property Tab

The Frame property tab allows the basic outline of the frame to be determined. From here frame duration is set, alone with the frame' s properties, and the frame type. The duration can be anywhere from self-paced to any amount of time desired measured in milliseconds. The type of multimedia desired can be selected under a Frame properties drop-down menu. Under the Frame type dropdown menu, the type of frame can be specified: Normal, Title, Review, Test, or Interaction.

2.2 Agent Property Tab

This is the general tab that is used to control the agent. Each frame can have up to three agent actions assigned to it. These actions are denoted as agent1, agent2, and agent3. However, these can be assigned as needed for example one agent can be given as many as three actions or three agents can be given one action each. These are selected from the available agents using the Agent dropdown menu. Just below this dropdown menu is a dropdown menu that specifies when the agent will be used. For example, "Action over frame" can be selected so the agent is active while the rest of the frame is running. Just below this are three additional tabs that specify (a) the agent's position on the screen, (b) what the agent will say in each frame, and (c) balloon formatting, if the agent has this option. These are the Action and Gesture tab, Speak and Play tab, and Balloon Setup tab, respectively.

Of these tabs, the Speak and Play tab is of the most importance. This frame in its most basic form allows for text to be entered into a text box. The agent reads this text using a text to speech engine. However, this text box can also contain simple markup within the text. This markup includes such speech parameters as volume, emphasis, pitch and speed. These markup tags can be inserted into the text by inputting the

BEST COPY AVAILABLE

desired values into the box beside the parameter name on the right portion of the tab and then double clicking the name. This list of parameters also includes a few special tags that can control the flow of the information delivery. These tags permit the agent to skip to a specific frame in the tree structure (Show Frame), or to go to specific frames in a selected Shockwave Flash movie (Go to Frame in Flash Movie). The remaining tag option is Insert Special action. This set of tags allows the user to start, stop, and restart a flash movie, and provides a tag that terminates the program at the end of a presentation. The Speak and Play tab allows for assignment of actions to the selected agent. These actions vary according to the abilities of the selected agent, and can be assigned either at the beginning or the end of the text the agent speaks.

Similar to other agent scripting tool, such as MASH, this agent property editor uses all available Microsoft agents controls. In addition, Xtrain utilizes the bookmark function of MS Agent to control the overall flow of the presentation. In fact it is the use of these bookmarking functions that make it possible to control Multimedia synchronization, such as with Flash animation, which is lacking in the other agent programs.

2.3 Text Display Tab

The text display tab is used to insert text to be displayed on the screen. Doing this involves clicking on the display area, typing in the text to be displayed, and then clicking update. The text will then appear in the display area in the same way that it will be displayed on the screen during the presentation.

2.4 Multimedia Tab

The Multimedia tab allows you to assign audio files, movie files, and wallpaper to the frame. The program supports wave files (.wav) and Enhanced Linguistic files audio formats. If an Enhanced Linguistic file is used Microsoft Agent can be made to appear to speak the file. The movie files available from this tab are AVI (.avi) and Mpeg (.mpg). A Bitmap (.bmp) image can be set as a background that either covers the whole screen or centered.

2.5 Pictures Tab

Using the picture tab, a picture can be added to the frame and manipulated. Xtrain supports two types of graphic files: Bitmap (.bmp) and GIF (.gif). The picture can be located at any point on the screen, centered, or can move from point to point. A hotspot option can be added to the picture to be used to give commands to the agent or to play audio files. Each hotspot can have information, such as text and tagged markup, to be sent to any selected agent.

2.6 Shockwave Tab

Under this tab, there are two options: Flash Movie and Shock Wave Movie. Flash movies and shockwave animations are among the most frequently used multimedia format. Xtrain uses activeX control from macromedia so both types of movies can be manipulated. By loading flash movie from this tab, detailed frame information can be examined so Agents can navigate through the movie. In addition, Xtrain uses FSCommand of flash movie to control Agent and the tree navigation.

2.7 Frame Summary Tab

The frame summary tab gives summary information both at the scripting phase and at the presentation phase. At the scripting phase, it gives a brief overview of the selections made in the other tabs for that frame. If the frame is a test frame, it also contains the correct answers to the questions given in the test frame. After viewing on the other half of this frame, responses are shown. If it was a test frame, the student' s responses are listed along with whether the response was correct.

2.8 HTML Tab

The program allows for the incorporation of html documents into presentations. This allows greater flexibility in terms of specialized displays. The format allows for html documents that are locally saved in the Xtrain directory to be displayed and navigated during presentations.

2.9 Test Tab

One other important feature of Xtrain is the testing option. During the scripting phase, frames can be assigned as testing frames on the frame property tab. These frames can be used to capture information from the user. They allow input in such forms as multiple-choice questions, short answer questions, and even essays. At the end of the presentation phase, input from the participant is automatically saved as an ASCII text file. The agent can also be programmed to give dynamic feedback, when the participant gives wrong answers.

3 Presentation Tool

The presentation of the scripted material is as easy as selecting the run drop-down menu and selecting the run entire session option. Alternatively, the Xtrain presentation file (.xtr) can be ran by double clicking its icon in the strain folder. This action occludes all other objects on the screen: only the scripted presentation and a control bar are visible. This control bar is a flash file that allows for the following actions: go back, continue, help, and progress. The presentation continues forward until it reaches the end of the presentation.

4 Summary

Xtrain is a program that is able to integrate multimedia files into one presentation format. The authoring side of the program takes advantage of many Windows' standards for ease of use. It provides a standard Windows interface window with icon buttons and drop-down menus, such as File, Edit, Window, and Help. These offer such options as open and save in the File menu, as well as, cut, copy, and paste in the Edit menu. Xtrain also offers a special drop down menu labeled Run. This menu offers the options of running the entire session or of previewing a selected frame. See Figure 1 for a view of the program. The frames are structured in a tree format that is located on the left of the screen. This tree is created via buttons labeled Brother, for frames on the same level, and Child, for frames on a branching level. Each frame can be scripted using nine different property tabs: Frame Property, Agent Property, Text Display, Multimedia, Picture, Shockwave, Frame Summary, HTML, and Test. These tabs may be individually associated with each frame. It is from these components that the script is produced to set the required tone for the information to be presented. Microsoft agent can also be used to control the flow between frames, so that if the need arises the agent can direct the presentation to any frame in the tree. Furthermore, if a Shockwave Flash file is used, the agent also has the ability to direct the flash movie to any frame within the movie. These options allow for maximum flexibility for the user when scripting a multimedia presentation. In addition to this freedom in scripting, Xtrain offers an easy presentation method that either selecting run entire session from the run menu or by simply double clicking on the created Xtrain file.

References

[1] Cassell, J., Bickmore, T., Billinghurst, M., Campbell, L., Chang, H., Vilhjalmsson, H., Yan, H., "Embodiment in Conversational Interfaces: Rea." Proceedings CHI' 99. Pittsburgh: ACM. (1999).

[2] Craig, S. D., & Hu, X. Xtrain: Expert training and data collection system. Presented at the 1999 meeting of The Society for Computers in Psychology. Los Angeles, California. (1999, November).

[3] Hu, X. "Xtrain 1.0 [computer program]." The University of Memphis: Author. (1998).

[4] Johnson, W. L., Rickel, J. W. & Lester, L. Animated pedagogical agents: Face-to-face interaction in interactive learning environments. International Journal of Artificial Intelligence in Education, Vol. 11, in press, (2000).

[5] Microsoft. Microsoft Agent 2.0 [computer program]. (1998).

[6] Mayer, R. E. Multimedia learning: Are we asking the right questions? Educational Psychologist, 32, 1-19, (1997).

[7] Mayer, R. Using Illustrations to Promote Constructivist Learning from Science Text. Psychology of Science Text Comprehension. (Submitted).

[8] Moreno, R., & Mayer, R. E. Cognitive principles of multimedia learning: The role of modality and contiguity. Journal of Educational Psychology, 91, 358-368, (1999).